ADD 404212

FZM-5567
15 May 1973

Contract No. F33615-69-C-1494

STRUCTURAL MATERIALS & DEVELOPMENT

ADVANCED COMP TE LO

ct

ANISOTROPIC CURVED PANEL ANALYSIS

D. J. Wilkins

Advanced Composites Division
Air Force Materials Laboratory
Wright-Patterson Air Force Base, Ohio

GENERAL DYNAMICS
Convair Aerospace Division

GENERAL DYNAMICS
Convair Aerospace Division
P. O. Box 748, Fort Worth, Texas 76101 • 817-732-4811
LG:kg/1494-FW#060-117

15 June 1973

Subject: Contract F33615-69-C-1494
Contract Compliance Submittal
General Dynamics Report FZM-5567

To: Air Force Materials Laboratory
Wright-Patterson Air Force Base, Ohio 45433

Attention: LC/Mr. W. R. Johnston
Contract F33615-69-C-1494
Item/Sub-Item IAA; B012

Inclosure: (A) Six (6) copies of General Dynamics Report FZM-5567 "Anisotropic Curved Panel Analysis", dated 15 May 1973

(B) Distribution List for General Dynamics Report FZM-5567

(C) Data Transmittal Sheet F1494-E-72, dated 14 June 1973

1. Inclosure (A) is submitted in compliance with Sequence No. B012 of Exhibit B, DD Form 1423 and Item IAA of Part I(A) of the Schedule to Contract F33615-69-C-1494.

2. Inclosure (B) is forwarded to show distribution of Inclosure (A) to other facilities.

3. It is requested that one (1) copy of Inclosure (C) be acknowledged and returned as specified.

GENERAL DYNAMICS
Convair Aerospace Division

H. J. Kauffman
Chief of Administrative Support

Direct transmittal is approved by the Air Force. Your answer, if any, will be forwarded through regular Air Force channels to General Dynamics, Convair Aerospace Division, Fort Worth, Texas.

cc: w/Incl's. (A) and (B)

AFSC
WPAFB-Ohio 45433
Attn: 4950/PMRB, Robert E. Gettier

cc:
AF (6)
TM
AFCMD/CMFCW
PD
P. D. Shockey, 2884

M. S. Howeth, 2860
M. D. Carter, 1128
A. Nole, 1214 (w/Incl. C)
L. G. Graham, 2236
Engr. Files, 2228
Corres. Control, 1137
Contract Records, 1135 (w/Incl. C)

Inclosure (B)
1494-FW#060-117

DISTRIBUTION LIST

	Addressee	No. of Copies
1.	Air Force Materials Laboratory Wright-Patterson AFB, Ohio 45433 Attn: Dr. Tsai/CA	1
2.	Air Force Materials Laboratory Wright-Patterson AFB, Ohio 45433 Attn: LAM (Technical Library)	1
3.	Air Force Materials Laboratory Wright-Patterson AFB, Ohio 45433 Attn: LNC/Mr. J. Ray	1
4.	Air Force Materials Laboratory Wright-Patterson Air Force Base, Ohio 45433 Attn: LN/M. H. Schwartz	1
5.	Air Force Materials Laboratory Wright-Patterson AFB, Ohio 45433 Attn: LTF/T. Cordell	1
6.	Air Force Flight Dynamics Laboratory Wright-Patterson AFB, Ohio 45433 Attn: FBC (Mr. P. A. Parmley)	2
7.	Air Force Flight Dynamics Laboratory Wright-Patterson AFB, Ohio 45433 Attn: FYS (Mr. M. Shirk)	1
8.	Hq. AFSC (DLTA) Andrews Air Force Base Washington, D. C. 20331 Attn: Dr. S. W. Strauss	1
9.	Aeronautical Systems Division Wright-Patterson AFB, Ohio 45433 Attn: XRHD/Mr. R. Stalder	1

Inclosure (B)
1494-FW#060-117

	Addressee	No. of Copies
10.	Aeronautical Systems Division Wright-Patterson AFB, Ohio 45433 Attn: ENFSR/Mr. D. W. Hinrichs	1
11.	Aeronautical Systems Division Wright-Patterson AFB, Ohio 45433 Attn: ENFS/Mr. G. Purkey	2
12.	Naval Air Development Center Johnsville, Warminster, Pa. 18974 Attn: AMFA-3/W. T. Miller	1
13.	Naval Air Engineering Center Philadelphia, Pa. 19101 Attn: ASL/Mr. R. Mollela	1
14.	Naval Air Development Center Aero Structures Department Code STD-8 Johnsville, Pennsylvania 18974 Attn: Mr. A. Manno	1
15.	Jet Propulsion Laboratory 4800 Oak Grove Drive Pasadena, California 91103 Attn: Mr. Jensen	1
16.	Department of Aeronautics and Astronautics Massachusetts Institute of Technology Cambridge, Mass. 02139 Attn: Prof. Theodore H. H. Pian Rm. 33-311	1
17.	Department of Mechanical Engineering Carnegi - Mellon Institute Pittsburgh, Penn. Attn: Dr. Tom Cruse	1

Inclosure (B)
1494-FW#060-117

	Addressee	No. of Copies
18.	National Aeronautics & Space Administration George C. Marshall Space Flight Center Huntsville, Alabama 35812 Attn: Erich E. Engler S&E - ASTN-ES	1
19.	NASA Langley Research Center Mail Stop 188 Hampton, Virginia 23365 Attn: Mr. R. Heldenfels	1
20.	SNSO - Cleveland Lewis Research Center Materials and Structures Branch 21000 Brookpark Road Cleveland, Ohio 44135 Attn: James J. Lombardo	1
21.	NASA - Lewis 21000 Brookpark Road Cleveland, Ohio 44135 Attn: C. Chamis Dept. MS-49	1
22.	Department of the Army Army Materials and Mechanics Research Center Watertown, Massachusetts 02172 Attn: Dr. E. M. Lenoe	1
23.	Department of the Navy Code 52032D Washington, D. C. 20360 Attn: M. Stander	1
24.	U.S. Naval Air Development Center Air Vehicle Technology Dept. Attn: Dr. E. J. McQuillan Code 3033 Warminster, Pa. 18974	1
25.	Aerospace Corporation P. O. Box 95085 Los Angeles, California 90045 Attn: D. E. Geiler Bldg. A2, Rm. 2223	1

Inclosure (B)
1494-FW#060-117

	Addressee	No. of Copies
26.	Bell Aerospace Corp. Structural Systems Department Buffalo, New York 14200 Attn: Mr. F. Anthony	1
27.	Bell Helicopter Co. P. O. Box 482 Fort Worth, Texas 76101 Attn: M. E. Glass	1
28.	The Boeing Company Missile and Information Systems Division P. O. Box 3999 Seattle, Washington 98124 Attn: Mr. J. Hoggatt	1
29.	The Boeing Company Airplane Division Seattle, Washington 98124 Attn: Dr. R. June	1
30.	Vertol Division Boeing Center Box 16858 Philadelphia, Pennsylvania 19142 Attn: Warren Stratton	1
31.	Fairchild-Hiller Corp. Republic Aviation Division Farmingdale, L.I., New York 11735 Attn: Dr. R. Levy	1
32.	Gates Learjet Corp. P. O. Box 1280 Wichita, Kansas 67201 Attn: Mr. Allyn M. Heinrich	1
33.	General Dynamics Convair Aerospace Division P. O. Box 1128 San Diego, California 92112 Attn: Mr. J. D. Forest	1

Inclosure (B)
1494-FW#060-117

	Addressee	No. of Copies
34.	General Electric Company P. O. Box 8555 Philadelphia, Penn. 19101 Attn: K. J. Hall Materials Performance Lab.	1
35.	Grumman Aircraft Corp. Plant #35, Advanced Composites Bethpage, L. I., New York 11714 Attn: Mr. R. N. Hadcock	1
36.	Hercules Powder Company P.O. Box 210 Cumberland, Maryland 21052 Attn: Mr. D. Hug	1
37.	Hercules, Inc. Bacchus Works P.O. Box 98 Magna, Utah 84044 Attn: Mr. H. R. MacPherson	1
38.	HITCO Materials Science Center 1600 W. 135th Street Gardena, California 90249 Attn: M. S. Allison	1
39.	ITT Research Institute 10 West 35th Street Chicago, Illinois 60616 Attn: Dr. R. Cornish	1

Inclosure (B)
1494-FW#060-117

	Addressee	No. of Copies
40.	Ling-Temco-Vought P. O. Box 5003 Dallas, Texas 75222 Attn: Mr. M. S. McClaren	1
41.	Lockheed Georgia Company Advanced Materials and Structural Division Dept. 72-26, Zone 459 Marietta, Georgia 30060 Attn: Mr. W. A. Pittman	1
42.	Lockheed Georgia Company Advanced Composites Info, CTR Marietta, Georgia 30060 Attn: Mr. W. G. Jurevic Dept. 72-14 Zone 402	1
43.	Lockheed - Georgia Company Marietta, Georgia Attn: J. E. Gilmer Dept. 42-11 Mail Zone 102	1
44.	Lockheed California P.O. Box 551 Burbank, California 91503 Attn: George Wald Dept. 74-50 Bldg. 243 Plant 2	1
45.	Lockheed Missile and Space Company Sunnyvale, California 94088 Attn: William Powell Department 81-12 Bldg. 154	1
46.	The Marquardt Corporation 16555 Saticoy Street Van Nuys, California 91406 Attn: Mr. B. A. Webb	1
47.	Art Feldman Mail Zone 1630 P.O. Box 179 Martin Denver Denver, Colorado 80201	1

Inclosure (B)
1494-FW#060-117

	Addressee	No. of Copies
48.	Material Sciences Corporation Blue Bell Office Campus 1777 Walton Road Blue Bell, Pennsylvania 19422 Attn: Carl Zweben	1
49.	McDonnell Aircraft Co. P. O. Box 516 St. Louis, Missouri 63166 Attn: McDonnell Library Dept. 218, Reference 70L1292	1
50.	McDonnell Aircraft Corp. Lambert-St. Louis Municipal Airport P. O. Box 516 St. Louis, Missouri 63166 Attn: Mr. R. C. Goran	1
51.	Minnesota Mining and Manufacturing Co. 1210 University Ave. St. Paul, Minnesota 55104 Attn: Mr. C. Jacobs	1
52.	Rockwell International Los Angeles Division International Airport Los Angeles, California 90009 Attn: Dr. L. M. Lackman	1
53.	Northrop Corporation Norair Division 3901 W. Broadway Hawthorne, California 90250 Attn: Mr. R. Hays	1
54.	Northrop Corporation Ventura Division 1515 Rancho Conejo Blvd. Newbury Park, California 91320 Attn: Dr. R. D. Johnson	1
55.	Southwest Research Institute 8500 Culerbra Road San Antonio, Texas 78206 Attn: Mr. G. C. Grimes	1

Inclosure (B)
1494-FW#060-117

	Addressee	No. of Copies
56.	Union Carbide Corporation Oakridge National Laboratory Isotopes Sales Department P. O. Box X Oakridge, Tennessee 37830 Attn: Mr. J. L. Cook	1
57.	Union Carbide Corporation Carbon Products Division P. O. Box 6116 Cleveland, Ohio 44101 Attn: J. Bowman	1
58.	U. S. Polymeric, Inc. 700 E. Dyer Road Santa Ana, California 92705 Attn: J. B. Williamson	1
59.	United Aircraft Corporation Hamilton Standard Composite Material Division Windsor Locks, Conn. 06096 Attn: Mr. Borie	1
60.	United Aircraft Research Labs. East Hartford, Connecticut 06108 Attn: Bruce Thompson	1
61.	Whittaker Corporation Narmco Materials Division 600 Victoria Street Costa Mesa, California 92626 Attn: H. King	1
62.	Whittaker Corporation Narmco R&D Division 3540 Aero Ct. San Diego, California 92123 Attn: Dr. A. Miller	1
63.	Skyline Industries P. O. Box 821 4901 NE Parkway Fort Worth, Texas 76101 Attn: T. P. Airhart Director Technical Products	1

Inclosure (B)
1494-FW#060-117

Addressee	No. of Copies
64. McDonnell Douglas Aircraft 3855 Lakewood Blvd. Long Beach, California 90801 Attn: W. D. Nelson Mail Stop C1-250 (35-41)	1
65. San Diego Aircraft Engineering 1730 Kettner Blvd. San Diego, California 92101 Attn: John Lyman	1
66. Plastics Technology Evaluation Center Picatinny Arsenal Dover, New Jersey 07801 Attn: SMUPA-VP3 Harry E. Pebley	1
67. Ryan Aeronautical Company Lindberg Field San Diego, California 92112 Attn: Mr. Roger Long	1
68. Marshall Space Flight Center S&E ASTN-MN B1.4612 Huntsville, Alabama 35812 Attn: J. E. Curry	1
69. Lockheed Missiles and Space Co. Dept. 81-12, Bldg. 154 P.O. Box 504 Sunnyvale, California 94088 Attn: Derek Yates	1

FZM-5567
15 May 1973

ANISOTROPIC CURVED PANEL ANALYSIS

Prepared by

Dr. D. J. Wilkins

Prepared for

Advanced Composites Division
Air Force Materials Laboratory
Air Force Systems Command
Wright-Patterson Air Force Base, Ohio

GENERAL DYNAMICS
Convair Aerospace Division
Fort Worth Operation

ABSTRACT

An analysis of laminated-composite cylindrically curved shells has been formulated and incorporated into digital computer procedure SS8. Many discrete effects were considered, including ring and stringer stiffening, by implementing a Rayleigh-Ritz energy analysis. The procedure solves static deflection, buckling, and natural frequency problems.

The results of an extensive experimental program for graphite-epoxy and boron-epoxy shells are included.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS
(Continued)

LIST OF ILLUSTRATIONS
(Continued)

LIST OF TABLES

LIST OF SYMBOLS

[A], [B], [D]	Constitutive matrix terms
A	Area
A_{rk}	Ring cross-sectional area, in^2.
$A_{s\ell}$	Stringer cross-sectional area, in^2.
a	Mode shape constants
a, b, h	Panel dimensions in (x;y;z) directions
C_{mj}	Mode shape constants
d	Strain energy partitions defined in Equations (14) - (22)
E_1	Fiber direction elastic modulus
E_2	Transverse direction elastic modulus
E_{rk}	Ring modulus of elasticity, psi.
$E_{s\ell}$	Stringer modulus of elasticity, psi.
f	Natural frequency, Hz.
G_{12}	In-plane shear modulus
$(GJ)_{rk}$	Ring torsional stiffness, $lb\text{-}in^2$.
$(GJ)_{s\ell}$	Stringer torsional stiffness, $lb\text{-}in^2$.
i_x, i_y	Initial modal term in (x;y) direction
I_{xxrk}	Moment of inertia of the ring area about the mid-surface x-axis at the line of attachment, $in.^4$
I_{xzrk}	Product of inertia of the ring area about the mid-surface x-z axis at the line of attachment, in^4
I_{zzrk}	Moment of inertia of the ring area about the z-axis at the line of attachment, $in.^4$

LIST OF SYMBOLS
(Continued)

$I_{yys\ell}$	Moment of inertia of the stringer area about the mid-surface y-axis at the line of attachment, in^4.
$I_{zzs\ell}$	Moment of inertia of the stringer area about the z-axis at the line of attachment, in^4.
$I_{yzs\ell}$	Product of inertia of the stringer area about the mid-surface y-z axis at the line of attachment, in^4.
K_L	Spring constant, lb/in/in
K_P	Spring constant lb/in
K_x, K_y, K_{xy}	Curvatures
K_{xy}	Proportionality constant (Tables VI and VII)
K_{x1}, K_{x2}, K_{y1}, K_{y2}	Rotational spring constants, in-lb/rad/in
M_L	Line moment, in-lb/in
M_P	Point moment, in-lb
M_x, M_y, M_{xy}	Moment resultants
m	axial mode number
$\bar{m}$	Lumped mass, lb-sec^2/in
N_x, N_y, N_{xy}	Stress resultants
n	circumferential mode number
n_x, n_y	Number of terms in (x;y) direction
P	Coefficients defined in Eqs. (30) - (32), lb/in.; load in ring or stringer, lb.
P	Pitch (Figure 17)
P_c	Point load, lb.

LIST OF SYMBOLS
(Continued)

Q	Potential energy of lateral loads
q	Coefficients defined in Eq. (42)
$\bar{q}$	Distributed lateral pressure
R	Radius
S	Linear part of U_p
T	Kinetic Energy
U	Potential energy of membrane loads
U_p	Total potential energy of membrane loads
u, v, w	Displacements in (x;y;z) direction
V	Potential energy
X, Y	Mode function in (x;y) direction
x, y, z	Coordinates in axial, circumferential, and radial directions, respectively.
x_k	Ring location
$\bar{x}_{rk}$	Location of ring centroid in the x-direction with respect to its line of attachment to the shell, in.
y_ℓ	Stringer location
$\bar{y}_{s\ell}$	Location of stringer centroid in the y-direction with respect to its line of attachment to the shell, in.
$\bar{z}_{rk}$	Location of ring centroid in the z-direction with respect to the middle surface of the shell at the line of attachment, in.
$\bar{z}_{s\ell}$	Location of stringer centroid in the z-direction with respect to the middle surface of the shell at the line of attachment, in.

LIST OF SYMBOLS (Continued)

α	Observation angle (Figure 17)
α_x, α_y	Constants defined by Equation (97)
β_x, β_y	Constants defined by Equation (97)
γ	Knockdown factor
δ	Distance defined in Figure 17
ϵ_x^o, ϵ_y^o, ϵ_{xy}^o	Midsurface strain
λ	Buckling eigenvalue
ν_{12}	Major Poisson's ratio
ρ	Density
ρ_{jm}	Mode shape functions
ρ_{rk}	Average density of ring material, $lb\text{-}sec^2/in^4$.
$\rho_{s\ell}$	Average density of stringer material, $lb\text{-}sec^2/in^4$.
τ	Time; shear stress
ϕ	Integrals defined by Equation (105)
ψ	Integrals defined by Equation (102)
Ω	Integrals defined by Equation (104)
ω	Circular frequency

SECTION I

INTRODUCTION

Modern aircraft are constructed with many curved panels. In the past, the use of isotropic materials permitted a relatively small number of tests to be used in the generation of simplified analytical methods and design curves. The advent of high-performance laminated composites has required the development of improved analysis tools since material properties of composites have defied simplification and their various coupling effects are often unconservative.

Ashton [1] has shown that the Rayleigh-Ritz method, when properly formulated and coupled with an efficient method of calculating the necessary integrals, can be a very versatile and efficient tool for structural analysis.

Consequently, an analysis tool for cylindrically curved anisotropic panels was proposed. The resulting program includes the following capabilities:

A. Types of Analysis

1. Static deflection and strength under complicated variations of edge and lateral loads with complicated support conditions

2. Elastic stability under complicated edge loads

3. Natural frequencies and mode shapes.

B. Geometry

1. Flat panel

2. Cylindrically curved panel

3. Full cylinder (specially orthotropic only).

C. Construction

1. Sheet with discrete rings and stringers

2. Sandwich with discrete rings and stringers (neglecting core shear).

D. Material - Linearly Elastic

1. Panel - layered anisotropic

2. Stiffeners - orthotropic.

E. Boundary Conditions

1. All combinations of clamped and simply supported; some combinations with free edges

2. Elastic moment restraint on opposite edges.

The analytical approach and the documentation of most of the required derivations is given in Section II. Other detailed derivations and assumptions are explained under the appropriate subroutine titles in the computer program documentation.

SECTION II

ANALYTICAL FORMULATION

2.1 METHOD OF ANALYSIS

The Rayleigh-Ritz energy method has been chosen for the analysis because of its versatility and speed when compared to finite-element or finite-difference techniques. Many effects may be considered by simply adding their contributions to the total energy of the system, without increasing the size of the resulting set of equations.

The basic energy principle involved is the theorem of stationary potential energy. In the present case it may be written as

$$V + U + Q - T = \text{constant} \tag{1}$$

where

V = strain energy

U = potential energy of membrane loads

Q = potential energy of lateral loads

T = kinetic energy

For a static deflection problem, Equation (1) takes the form

$$V + U + Q = \text{constant} \tag{2}$$

For an elastic stability problem, Equation (1) becomes

$$V + \lambda U = \text{constant} \tag{3}$$

where λ is the buckling eigenvalue.

For a free-vibration problem, including membrane loads, Equation (1) is reduced to

$$V + U - T = \text{constant} \tag{4}$$

These energies are formulated in the following sections. The Rayleigh-Ritz method is then applied to form a set of simultaneous equations for the static deflection problem, or a standard eigenvalue problem for the buckling and vibration cases. This resulting problem is solved with a digital computer program, as described in Appendix I.

All of the following assumptions will be implicit in the analysis:

1. The shell is thin and has constant thickness
2. The displacements are small when compared to the thickness
3. Transverse shear effects are negligible.

2.2 RAYLEIGH-RITZ METHOD

As noted above, each of the problems of concern is governed by Equation (1), where the variations can be replaced with the problem of finding the minimum of Equation (1) by assuming the displacements in the form of a finite series:

$$
\begin{aligned}
u &= \sum_{m=m_i}^{m_f} \sum_{n=n_i}^{n_f} a_{1mn} X_{1m}(x) Y_{1n}(y) \sin \omega \tau \\
v &= \sum_{m=m_i}^{m_f} \sum_{n=n_i}^{n_f} a_{2mn} X_{2m}(x) Y_{2n}(y) \sin \omega \tau \\
w &= \sum_{m=m_i}^{m_f} \sum_{n=n_i}^{n_f} a_{3mn} X_{3m}(x) Y_{3n}(y) \sin \omega \tau
\end{aligned} \tag{5}
$$

where

$$m_i = i_x \quad ; \quad m_f = i_x + n_x - 1$$

$$n_i = i_y \quad ; \quad n_f = i_y + n_y - 1$$

the a_{imn} are undetermined constants, and the functions X_{im}, Y_{in} are chosen to satisfy the geometric boundary conditions on u, v, and w. Introducing the assumed series into Equation (1) reduces the problem to finding the minimum of Equation (1) with respect to the undetermined constants, a_{imn}. Thus, Equation (1) is now a function of only the undetermined constants, a_{imn}, and is equivalent to the following conditions:

$$\frac{\partial}{\partial a_{imn}}(V+U+Q-T) = 0 \qquad (6)$$

where $i = 1, 2, 3$; $m = m_i \ldots, m_f$; $n = n_i \ldots, n_f$; such that Equation (6) denotes a set of $3\, n_x n_y$ simultaneous algebraic equations, for which solution techniques are readily available.

The assumed series (5) always involve additional constraints on the energy criteria beyond the physical constraints on the problem, so that the solution obtained by the Rayleigh-Ritz method is always in the direction of a stiffer structure. However, if the assumed series is complete and satisfies the geometric boundary conditions, then the consecutive solutions obtained by including additional terms in the assumed series must approach the correct solution.

2.3 SHELL THEORY

Before proceeding with the analysis, a set of equations defining the midsurface strains and curvatures in terms of the deflections u, v, and w are required. These strain-displacement relations constitute the shell theory being used. Several theories are commonly used, namely Love's, Donnell's, Novozhilov's, etc. In this work, Vlasov [2] shell theory will be used. In the present notation, it requires that

$$\begin{aligned}
\epsilon_x^o &= u_{,x} \\
\epsilon_y^o &= v_{,y} + w/R \\
\epsilon_{xy}^o &= u_{,y} + v_{,x} \\
K_x &= -w_{,xx} \\
K_y &= -w_{,yy} - R^{-2} w \\
K_{xy} &= -2 w_{,xy} - R^{-1} u_{,y} + R^{-1} v_{,x}
\end{aligned} \qquad (7)$$

where the commas denote partial differentiation with respect to the variables following them; the coordinate system and sign conventions are shown in Figures 1 and 2.

With the definitions of Equation (7), the total strain at any point at a distance z from the middle surface is written as

$$\begin{aligned} \epsilon_x &= \epsilon_x^o + z K_x \\ \epsilon_y &= \epsilon_y^o + z K_y \\ \epsilon_{xy} &= \epsilon_{xy}^o + z K_{xy} \end{aligned} \tag{8}$$

2.4 SHELL STRAIN ENERGY

The derivation of the shell strain energy is necessary for all three of the analyses to be performed. The derivation depends on the coordinate system and sign conventions shown in Figures 1 and 2.

For a laminated anisotropic material, the constitutive relations [3] are

$$\begin{bmatrix} N_x \\ N_y \\ N_{xy} \\ M_x \\ M_y \\ M_{xy} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{16} & B_{11} & B_{12} & B_{16} \\ A_{12} & A_{22} & A_{26} & B_{12} & B_{22} & B_{26} \\ A_{16} & A_{26} & A_{66} & B_{16} & B_{26} & B_{66} \\ B_{11} & B_{12} & B_{16} & D_{11} & D_{12} & D_{16} \\ B_{12} & B_{22} & B_{26} & D_{12} & D_{22} & D_{26} \\ B_{16} & B_{26} & B_{66} & D_{16} & D_{26} & D_{66} \end{bmatrix} \begin{bmatrix} \epsilon_x^o \\ \epsilon_y^o \\ \epsilon_{xy}^o \\ K_x \\ K_y \\ K_{xy} \end{bmatrix} \tag{9}$$

which includes bending-stretching coupling, as well as coupling between normal stress, shearing and twisting deformations.

The strain energy of the shell may be concisely stated as

SMD7037

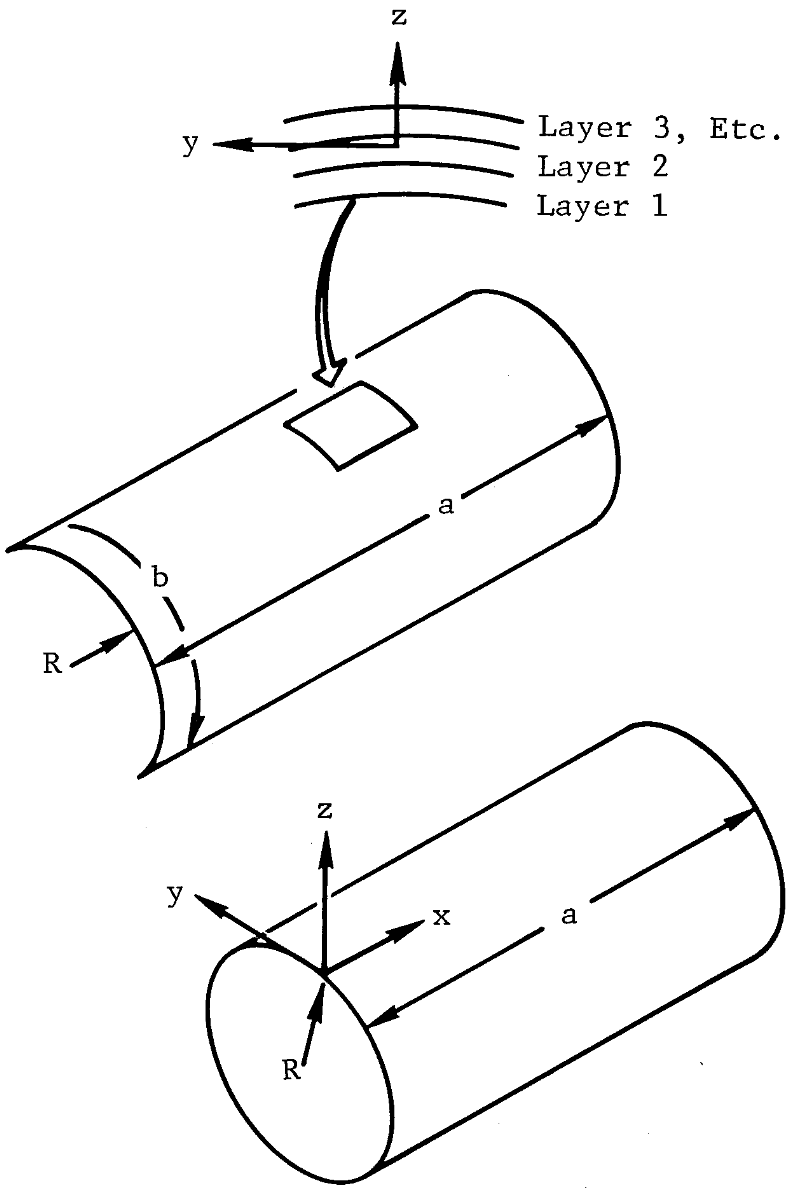


Figure 1 Shell Geometry

SMD7038

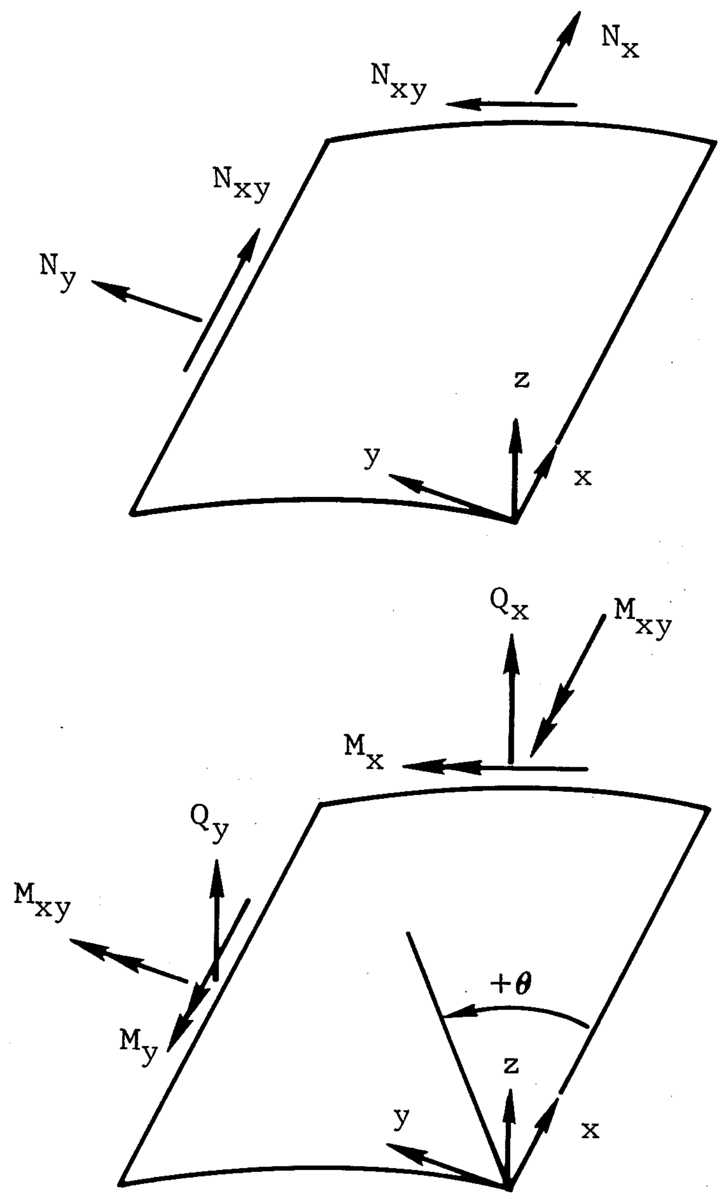


Figure 2 Sign Convention for Positive Loads

$$V_s = \frac{1}{2} \iint_A \begin{Bmatrix} N \\ M \end{Bmatrix}^T \begin{Bmatrix} \epsilon^\circ \\ K \end{Bmatrix} dA \tag{10}$$

which, after substituting from Equation (9), takes the form

$$V_s = \frac{1}{2} \iint_{Area} \{\epsilon^\circ\}^T [A] \{\epsilon^\circ\} + 2\{\epsilon^\circ\}^T [B] \{K\} + \{K\}^T [D] \{K\} \, dA \tag{11}$$

Using Equations (7) and (9) and performing the indicated matrix operations in Equation (11) results in the following:

$$\begin{aligned}
V_s = \frac{1}{2} \iint_A & A_{11} [u_{,x}^2] + 2A_{12} [u_{,x} u_{,y} + R^{-1} u_{,x} w] \\
& + 2A_{16} [u_{,x} u_{,y} + u_{,x} v_{,x}] + A_{22} [v_{,y} + R^{-1} w]^2 \\
& + 2A_{26} [u_{,y} v_{,y} + v_{,x} v_{,y} + R^{-1} u_{,y} w + R^{-1} v_{,x} w] + A_{66} [u_{,y} + v_{,x}]^2 \\
& - 2B_{11} [u_{,x} w_{,xx}] - 2B_{12} [v_{,y} w_{,xx} + R^{-1} w w_{,xx} + u_{,x} w_{,yy} - R^{-2} u_{,x} w] \\
& + 2B_{16} [R^{-1} u_{,x} v_{,x} - u_{,y} w_{,xx} - v_{,x} w_{,xx} - 2 u_{,x} w_{,xy} - R^{-1} u_{,x} u_{,y}] \\
& - 2B_{22} [v_{,y} w_{,yy} + R^{-1} w w_{,yy} + R^{-2} v_{,y} w + R^{-3} w^2] \\
& + 2B_{26} [R^{-1} v_{,x} v_{,y} - u_{,y} w_{,yy} - v_{,x} w_{,yy} - 2 v_{,y} w_{,xy} - 2 R^{-1} w w_{,xy} \\
& - R^{-1} u_{,y} v_{,y} - 2 R^{-2} u_{,y} w] + 2 B_{66} [R^{-1} v_{,x}^2 - 2 u_{,y} w_{,xy} - 2 v_{,x} w_{,xy} \\
& - R^{-1} u_{,y}^2] + D_{11} [w_{,xx}^2] + 2D_{12} [w_{,xx} w_{,yy} + R^{-2} w w_{,xx}] \\
& + 2D_{16} [2 w_{,xx} w_{,xy} + R^{-1} u_{,y} w_{,xx} - R^{-1} v_{,x} w_{,xx}]
\end{aligned} \tag{12}$$

$$+ D_{22}\left[2R^{-2}w\,w_{,yy} + R^{-4}w^2 + w_{,yy}^2\right] + 2D_{26}\left[2w_{,xy}w_{,yy}\right.$$

$$\left. + R^{-1}u_{,y}w_{,yy} - R^{-1}v_{,x}w_{,yy} + 2R^{-2}w\,w_{,xy} + R^{-3}u_{,y}w - R^{-3}v_{,x}w\right]$$

$$+ D_{66}\left[4R^{-1}u_{,y}w_{,xy} - 4R^{-1}v_{,x}w_{,xy} - 2R^{-2}u_{,y}v_{,x} + R^{-2}u_{,y}^2\right.$$

$$\left. + R^{-2}v_{,x}^2 + 4w_{,xy}^2\right] \quad d\,(\text{Area})$$

Substitution of Equation (5) into Equation (12), nondimensionalization of the shape functions, taking partial derivatives with respect to the undetermined constants, and defining the integral functions ψ gives

$$\frac{\partial V_s}{\partial a_{kij}} = \sum_{\ell=1}^{3} \sum_{m=m_i}^{m_f} \sum_{n=n_i}^{n_f} d_{k\ell ijmn}\, a_{\ell mn} \qquad \begin{cases} k = 1,2,3 \\ i = m_i, \ldots, m_f \\ j = n_i, \ldots, n_f \end{cases} \tag{13}$$

where

$$\begin{aligned} d_{11ijmn} = {} & A_{11}a^{-1}b\left[\psi_{x21i1m}\,\psi_{y11j1n}\right] + (A_{16} - B_{16}R^{-1})\left[\psi_{x41i1m}\,\cdot\right. \\ & \left.\psi_{y41m1j} + \psi_{x41m1i}\,\psi_{y41j1n}\right] + ab^{-1}(A_{66} - 2B_{66}R^{-1} \\ & + D_{66}R^{-2})\left[\psi_{x11i1m}\,\psi_{y21j1n}\right] \end{aligned} \tag{14}$$

$$\begin{aligned} d_{12ijmn} = {} & A_{12}\left[\psi_{x41i2m}\,\psi_{y42n1j}\right] + ab^{-1}(A_{16} \\ & + B_{16}R^{-1})\left[\psi_{x21i2m}\,\psi_{y11j2n}\right] + ab^{-1}(A_{26} + B_{26}R^{-1}) \cdot \\ & \left[\psi_{x11i2m}\,\psi_{y21j2n}\right] + (A_{66} + B_{66}R^{-1})\left[\psi_{x42m1i}\,\psi_{y41j2n}\right] \end{aligned} \tag{15}$$

$$
\begin{aligned}
d_{13ijmn} = {} & bR^{-1}(A_{12} - R^{-1}B_{12})[\psi_{x41i3m}\psi_{y11j3n}] \\
& + aR^{-1}(A_{26} - 2R^{-1}B_{26} + R^{-2}D_{26})[\psi_{x11i3m}\psi_{y41j3n}] \\
& - B_{11}a^{-2}b[\psi_{x63m1i}\psi_{y11j3n}] - B_{12}b^{-1}[\psi_{x41i3m}\psi_{y53n1j}] \\
& - 2B_{16}a^{-1}[\psi_{x21i3m}\psi_{y43n1j}] + a^{-1}(D_{16}R^{-1} - B_{16})[\psi_{x53m1i} \cdot \\
& \psi_{y41j3n}] + ab^{-2}(D_{26}R^{-1} - B_{26})[\psi_{x11i3m}\psi_{y63n1j}] \\
& + 2b^{-1}(D_{66}R^{-1} - B_{66})[\psi_{x43m1i}\psi_{y21j3n}]
\end{aligned} \tag{16}
$$

$$
\begin{aligned}
d_{22ijmn} = {} & A_{22}ab^{-1}[\psi_{x12i2m}\psi_{y22j2n}] + (A_{26} + R^{-1}B_{26}) \cdot \\
& [\psi_{x42i2m}\psi_{y42n2j} + \psi_{x42m2i}\psi_{y42j2n}] + a^{-1}b \cdot \\
& (A_{66} + 2R^{-1}B_{66} + R^{-2}D_{66})[\psi_{x22i2m}\psi_{y12j2n}]
\end{aligned} \tag{17}
$$

$$
\begin{aligned}
d_{23ijmn} = {} & aR^{-1}(A_{22} - R^{-1}B_{22})[\psi_{x12i3m}\psi_{y42j3n}] \\
& - a^{-1}B_{12}[\psi_{x53m2i}\psi_{y42j3n}] - a^{-2}b(B_{16} + R^{-1}D_{16}) \cdot \\
& [\psi_{x63m2i}\psi_{y12j3n}] + bR^{-1}(A_{26} - R^{-2}D_{26}) \cdot \\
& [\psi_{x42i3m}\psi_{y12j3n}] - ab^{-2}B_{22}[\psi_{x12i3m}\psi_{y63n2j}] \\
& - 2b^{-1}B_{26}[\psi_{x43m2i}\psi_{y22j3n}] - b^{-1}(B_{26} + R^{-1}D_{26}) \cdot \\
& [\psi_{x42i3m}\psi_{y53n2j}] - 2a^{-1}(B_{66} + R^{-1}D_{66})[\psi_{x22i3m}\psi_{y43n2j}]
\end{aligned} \tag{18}
$$

$$
\begin{aligned}
d_{33ijmn} = {} & abR^{-2}(A_{22} - 2B_{22}R^{-1} + D_{22}R^{-2})[\psi_{x13i3m}\psi_{y13j3n}] \\
& + a^{-1}bR^{-1}(D_{12}R^{-1} - B_{12})[\psi_{x53i3m}\psi_{y13j3n} + \psi_{x53m3i}\psi_{y13j3n}] \\
& + ab^{-1}R^{-1}(D_{22}R^{-1} - B_{22})[\psi_{x13i3m}\psi_{y53j3n} + \psi_{x13i3m}\psi_{y53n3j}] \\
& + 2R^{-1}(D_{26}R^{-1} - B_{26})[\psi_{x43i3m}\psi_{y43j3n} + \psi_{x43m3i}\psi_{y43n3j}] \\
& + a^{-3}bD_{11}[\psi_{x33i3m}\psi_{y13j3n}] \\
& + a^{-1}b^{-1}D_{12}[\psi_{x53i3m}\psi_{y53n3j} + \psi_{x53m3i}\psi_{y53j3n}] \\
& + 2a^{-2}D_{16}[\psi_{x63i3m}\psi_{y43n3j} + \psi_{x63m3i}\psi_{y43j3n}] \\
& + ab^{-3}D_{22}[\psi_{x13i3m}\psi_{y33j3n}] \\
& + 2b^{-2}D_{26}[\psi_{x43i3m}\psi_{y63n3j} + \psi_{x43m3i}\psi_{y63j3n}] \\
& + 4a^{-1}b^{-1}D_{66}[\psi_{x23i3m}\psi_{y23j3n}]
\end{aligned}
\tag{19}
$$

The integral functions ψ are defined and explained in Section 2.10. Note also that

$$d_{21ijmn} = d_{12mnij} \tag{20}$$

$$d_{31ijmn} = d_{13mnij} \tag{21}$$

$$d_{32ijmn} = d_{23mnij} \tag{22}$$

so that the potential energy matrix is symmetric.

2.5 SHELL KINETIC ENERGY

The kinetic energy of the vibrating shell is based on the translational inertia in the three coordinate directions. The rotatory inertia components are neglected to maintain consistency with the previous deletion of transverse shear flexibilities. The mass times velocity-squared is written on a differential basis as

$$T = \frac{1}{2}\rho \int_z \int_0^b \int_0^a (u_{,\tau}^2 + v_{,\tau}^2 + w_{,\tau}^2)\, dx\, dy\, dz \qquad (23)$$

The integral through the thickness is trivial, giving

$$T = \frac{1}{2}\rho h \int_0^b \int_0^a (u_{,\tau}^2 + v_{,\tau}^2 + w_{,\tau}^2)\, dx\, dy \qquad (24)$$

Performing the same substitution of the assumed modes, taking partials with respect to the undetermined constants, and using the integral definitions as for the potential energy derivations, results in the following required expressions for the variations of the kinetic energy:

$$\frac{\partial T}{\partial a_{1ij}} = \rho h a b \omega^2 \sum_m \sum_n \psi_{x11im} \psi_{y1j1jm} a_{1mn} \qquad (25)$$

$$\frac{\partial T}{\partial a_{2ij}} = \rho h a b \omega^2 \sum_m \sum_n \psi_{x12i2m} \psi_{y12j2n} a_{2mn} \qquad (26)$$

$$\frac{\partial T}{\partial a_{3ij}} = \rho h a b \omega^2 \sum_m \sum_n \psi_{x13i3m} \psi_{y13j3n} a_{3mn} \qquad (27)$$

2.6 POTENTIAL ENERGY OF INPLANE LOADS

The total potential energy of the inplane loads on a panel may be simply formed as the product of the vector of running loads and the vector of mid-plane strains:

$$U_p = -\iint_A \{N\}^T \{\epsilon^\circ\} \, dA \tag{28}$$

Expanding the strains according to Equation (7) and including first-order nonlinear terms results in

$$U_p = -\iint_A \Big\{ N_x \left[u_{,x} + \tfrac{1}{2} w_{,x}^2\right] + N_y \left[v_{,y} + \bar{R}^{-1} w + \tfrac{1}{2} w_{,y}^2\right] + N_{xy} \left[v_{,x} + u_{,y} + w_{,x} w_{,y}\right] \Big\} \, dA \tag{29}$$

To allow integration of Equation (29) requires an assumed form for N_x, N_y, and N_{xy}. The form assumed here is a power series in the x and y directions, defined as

$$N_x = \sum_{k=1}^{10} \sum_{\ell=1}^{10} P_{xk\ell} \left(\frac{x}{a}\right)^{k-1} \left(\frac{y}{b}\right)^{\ell-1} \tag{30}$$

$$N_y = \sum_{k=1}^{10} \sum_{\ell=1}^{10} P_{yk\ell} \left(\frac{x}{a}\right)^{k-1} \left(\frac{y}{b}\right)^{\ell-1} \tag{31}$$

$$N_{xy} = \sum_{k=1}^{10} \sum_{\ell=1}^{10} P_{xyk\ell} \left(\frac{x}{a}\right)^{k-1} \left(\frac{y}{b}\right)^{\ell-1} \tag{32}$$

Before integrating Equation (29), U_p is separated into its linear and nonlinear terms,

$$U_p = S + U \tag{33}$$

where the linear terms are retained in S and the nonlinear terms are retained in U. Then,

$$S = -\iint \sum_k \sum_l \left\{ P_{xkl}[u_{,x}] + P_{ykl}[v_{,y} + W/R] + P_{xykl}[v_{,x} + u_{,y}] \right\} \left(\frac{x}{a}\right)^{k-1} \left(\frac{y}{b}\right)^{l-1} dA \tag{34}$$

Using the definitions of u, v, and w from Equation (5),

$$\begin{aligned} S = -\iint \sum_k \sum_l \Big\{ & P_{xkl}\Big[\bar{a}^{1} \sum_m \sum_n X_{1m,x} Y_{1n}\, a_{1mn}\Big] \\ & + P_{ykl}\Big[b^{-1} \sum_m \sum_n X_{2m} Y_{2n,y}\, a_{2mn} + R^{-1} \sum_m \sum_n X_{3m} Y_{3n} a_{3mn}\Big] \\ & + P_{xykl}\Big[\bar{a}^{1} \sum_m \sum_n X_{2m,x} Y_{2n} a_{2mn} + b^{-1} \sum_m \sum_n X_{1m} Y_{1n,y} \cdot \\ & a_{1mn}\Big]\Big\} \left(\frac{x}{a}\right)^{k-1} \left(\frac{y}{b}\right)^{l-1} dA \end{aligned} \tag{35}$$

Taking partials with respect to the coefficients and using the integral definitions of Section 2.10 gives

$$\begin{aligned} \frac{\partial S}{\partial a_{1ij}} &= -\sum_k \sum_l \left\{ P_{xkl}\left[b\, \Phi_{kx21i}\, \Phi_{ly11j}\right] + P_{xykl}\left[a\, \Phi_{kx11i}\, \Phi_{ly21j}\right] \right\} \\ \frac{\partial S}{\partial a_{2ij}} &= -\sum_k \sum_l \left\{ P_{ykl}\left[a\, \Phi_{kx12i}\, \Phi_{ly22j}\right] + P_{xykl}\left[b\, \Phi_{kx22i}\, \Phi_{ky12j}\right] \right\} \\ \frac{\partial S}{\partial a_{3ij}} &= -\sum_k \sum_l \left\{ P_{xykl}\left[ab R^{-1} \Phi_{kx13i}\, \Phi_{ly13j}\right] \right\} \end{aligned} \tag{36}$$

Similarly for U,

$$U = -\iint \sum_{k} \sum_{\ell} \left\{ P_{xk\ell}\left[\tfrac{1}{2} w_{,x}^{2}\right] + P_{yk\ell}\left[\tfrac{1}{2} w_{,y}^{2}\right] + P_{xyk\ell}\left[w_{,x} w_{,y}\right] \right\} \left(\tfrac{x}{a}\right)^{k-1} \left(\tfrac{y}{b}\right)^{\ell-1} dA \tag{37}$$

Substituting in the definitions of Equation (5), taking partial derivatives and using the integral definitions of Section 2.10 gives

$$\frac{\partial U}{\partial a_{1ij}} = \frac{\partial U}{\partial a_{2ij}} = 0$$

$$\begin{aligned} \frac{\partial U}{\partial a_{3ij}} = -\sum_{m} \sum_{n} \sum_{k} \sum_{\ell} \Big[& a^{-1} b P_{xk\ell}\left(\Omega_{k12im}\Omega_{\ell 2ijn}\right) \\ & + a b^{-1} P_{yk\ell}\left(\Omega_{k11im}\Omega_{\ell 22jn}\right) + P_{xyk\ell} \cdot \\ & \left(\Omega_{k13im}\Omega_{\ell 23nj} + \Omega_{k13mi}\Omega_{\ell 23jn}\right)\Big] a_{3mn} \end{aligned} \tag{38}$$

2.7 POTENTIAL ENERGY OF LATERAL LOADS

A distributed lateral pressure is defined by power series in the x and y directions as

$$\bar{q} = \sum_{k=1}^{10} \sum_{\ell=1}^{10} q_{k\ell} \left(\tfrac{x}{a}\right)^{k-1} \left(\tfrac{y}{b}\right)^{\ell-1} \tag{39}$$

The potential energy of this load is

$$Q = \iint_{A} \bar{q}\, w \, dA \tag{40}$$

Combination of the definitions for $\bar{q}$ and w, differentiation with respect to the coefficients, and use of the integral definitions results in

$$\frac{\partial Q}{\partial a_{1ij}} = \frac{\partial Q}{\partial a_{2ij}} = 0$$

$$\frac{\partial Q}{\partial a_{3ij}} = \sum_k \sum_l a b g_{kl} \Phi_{kx13i} \Phi_{ly13j} \tag{41}$$

2.8 DISCRETE ENERGY CONTRIBUTIONS

As noted above, a significant reason for employing the Rayleigh-Ritz energy method is the ease with which many desired effects may be included. These effects and their required energy formulations are described below.

2.8.1 Stiffeners

An important effect to be included for aircraft curved panels is that of discrete, eccentric stiffening elements. These are called stringers in the x-direction and rings in the y-direction.

2.8.1.1 Stringers

The energy contributions for the discrete, eccentric stringers were adapted from Reference [4]. The appropriate geometry for the stiffened shell and the stiffeners themselves is shown in Figures 3 and 4. The potential energy of the stringers due to extension, bending, and torsion, neglecting the bending-torsion coupling, is expressed

$$\begin{aligned}\Delta V = \sum_{l=1}^{L} \frac{E_{sl}}{2} \int_0^a \Big[& \big(A_{sl} u_{,x}^2 - 2\bar{y}_{sl} A_{sl} u_{,x} v_{,xx} \\ & + I_{zzsl} v_{,xx}^2\big) + I_{yysl} w_{,xx}^2 - 2\bar{z}_{sl} A_{sl} u_{,x} w_{,xx} \\ & + 2 I_{yzsl} v_{,xx} w_{,xx} \Big]_{y=y_l} dx + \frac{(GJ)_{sl}}{2} \int_0^a w_{,xy}^2 \Big|_{y=y_l} dx\end{aligned} \tag{42}$$

SMD3102

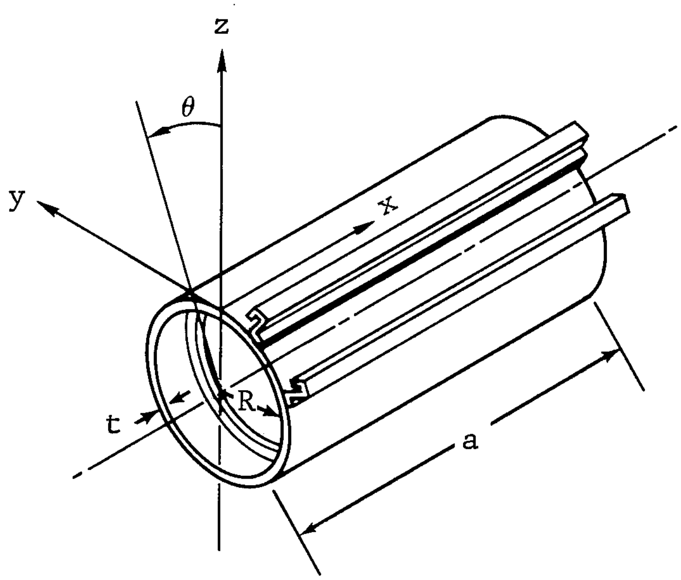
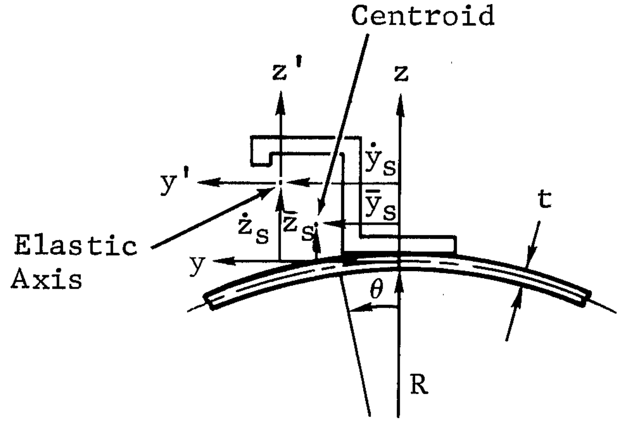
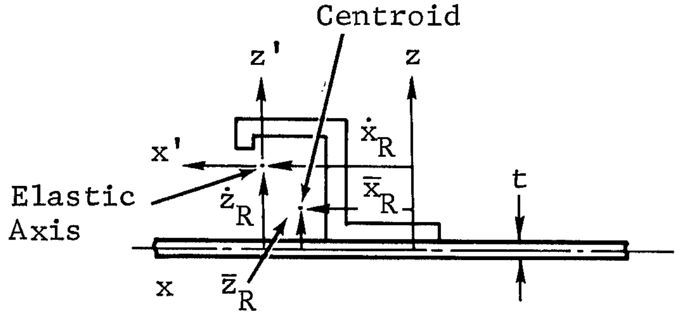


Figure 3 Geometry of Discretely Stiffened Cylinder

SMD3139



External Stringer Detail



External Ring Detail

Figure 4 Geometric Detail of Eccentric Stiffeners

The form of the partials after introduction of the assumed modes, nondimensionalization, and integration is

$$\frac{\partial \Delta V}{\partial a_{1ij}} = \sum_{l=1}^{L} \sum_{m} \sum_{n} E_{sl} A_{sl} \Big[a^{-1} \psi_{x21i1m} Y_{1j} Y_{1n} a_{1mn} - a^{-2} \bar{y}_{sl} \psi_{x62m1i} Y_{1j} Y_{2n} a_{2mn} - a^{-2} \bar{z}_{sl} \psi_{x63m1i} \cdot Y_{1j} Y_{3n} a_{3mn} \Big]_{y=y_l} \tag{43}$$

$$\frac{\partial \Delta V}{\partial a_{2ij}} = \sum_{l=1}^{L} \sum_{m} \sum_{n} E_{sl} \Big[-a^{-2} \bar{y}_{sl} \psi_{x62i1m} Y_{2j} Y_{1n} a_{1mn} + a^{-3} I_{zzsl} \psi_{x32i2m} Y_{2j} Y_{2n} a_{2mn} + a^{-3} I_{yzsl} \cdot \psi_{x32i3m} Y_{2j} Y_{3n} a_{3mn} \Big]_{y=y_l} \tag{44}$$

$$\frac{\partial \Delta V}{\partial a_{3ij}} = \sum_{l=1}^{L} \sum_{m} \sum_{n} \Big\{ E_{sl} \Big[a^{-3} I_{yysl} \psi_{x33i3m} Y_{3j} Y_{3n} a_{3mn} - a^{-2} \bar{z}_{sl} A_{sl} \psi_{x63i1m} Y_{3j} Y_{1n} a_{1mn} + a^{-3} I_{yzsl} \cdot \psi_{x33i2m} Y_{3j} Y_{2n} a_{2mn} \Big] + a b^{-2} (GJ)_{sl} \Big[\psi_{x23i3m} Y_{3j,y} Y_{3n,y} a_{3mn} \Big] \Big\}_{y=y_l} \tag{45}$$

The stringer kinetic energy is expressed by

$$\Delta T = \tfrac{1}{2} \sum_{l=1}^{L} \rho_{sl} \int_0^a \Big[A_{sl} \big(u_{,\tau}^2 - 2 \bar{y}_{sl} u_{,\tau} v_{,x\tau} + v_{,\tau}^2 - 2 \bar{z}_{sl} u_{,\tau} w_{,x\tau} - 2 \bar{z}_{sl} v_{,\tau} w_{,y\tau} + w_{,\tau}^2 + 2 \bar{y}_{sl} w_{,\tau} w_{,y\tau} \big) \tag{46}$$

$$+ I_{zzs\ell}\left(v_{,x\tau}^2 + w_{,y\tau}^2\right) + 2 I_{yzs\ell}\left(v_{,x\tau} w_{,x\tau}\right)$$

$$+ I_{yys\ell}\left(w_{,x\tau}^2 + w_{,y\tau}^2\right)\Big]_{y=y_\ell} dx \qquad \text{(46) Cont'd.}$$

In final form, the partials of the stringer kinetic energy are expressed as

$$\frac{\partial \Delta T}{\partial a_{1ij}} = \sum_{\ell=1}^{L} \sum_m \sum_n \rho_{s\ell}\,\omega^2 A_{s\ell}\Big[a\,\psi_{x11i1m} Y_{1j} Y_{1n} a_{1mn}$$
$$- \bar{y}_{s\ell}\,\psi_{x42m1i} Y_{1j} Y_{2n} a_{2mn} - \bar{z}_{s\ell}\,\psi_{x43m1i} Y_{1j} Y_{3n} a_{3mn}\Big]_{y=y_\ell} \qquad (47)$$

$$\frac{\partial \Delta T}{\partial a_{2ij}} = \sum_{\ell=1}^{L} \sum_m \sum_n \rho_{s\ell}\,\omega^2 \Big\{\Big[-A_{s\ell}\,\bar{y}_{s\ell}\,\psi_{x42i1m} Y_{2j} Y_{1n}\Big] a_{1mn}$$
$$+ \Big[a A_{s\ell}\,\psi_{x12i2m} + \bar{a}^1 I_{zzs\ell}\,\psi_{x22i2m}\Big] Y_{2j} Y_{2n} a_{2mn} \qquad (48)$$
$$+ \Big[\bar{a}^1 I_{yys\ell}\,\psi_{x22i3m} Y_{2j} Y_{3n} - a\bar{b}^{-1} \bar{z}_{s\ell} A_{s\ell}\,\psi_{x12i3m} Y_{2j} \cdot$$
$$Y_{3n,y}\Big] a_{3mn}\Big\}_{y=y_\ell}$$

$$\frac{\partial \Delta T}{\partial a_{3ij}} = \sum_{\ell=1}^{L} \sum_m \sum_n \rho_{s\ell}\,\omega^2 \Big\{\Big[-\bar{z}_{s\ell} A_{s\ell}\,\psi_{x43i1m} Y_{3j} Y_{1n}\Big] \cdot$$
$$a_{1mn} + \Big[-\bar{z}_{s\ell}\, a\bar{b}^{-1} A_{s\ell}\,\psi_{x13i2m} Y_{3j,y} + \bar{a}^1 I_{yzs\ell} \cdot$$
$$\psi_{x23i2m} Y_{3j}\Big] Y_{2n} a_{2mn} + \Big[\left(\psi_{x13i3m}\right)\Big(a A_{s\ell} Y_{3j} \cdot \qquad (49)$$
$$Y_{3n} + a\bar{b}^{-1} \bar{y}_{s\ell} A_{s\ell}\left[Y_{3j} Y_{3n,y} + Y_{3j,y} Y_{3n}\right] + a\bar{b}^{-2}\, Y_{3j,y} \cdot$$
$$Y_{3n,y}\left[I_{zzs\ell} + I_{yys\ell}\right]\Big) + \bar{a}^1 I_{yys\ell}\,\psi_{x23i3m} Y_{3j} Y_{3n}\Big] a_{3mn}\Big\}_{y=y_\ell}$$

The remaining stringer energy contribution arises from an external axial tension load, P_x, on the stringer. This energy is written

$$\Delta U = -\int_x P_x \left[u_{,x} - \bar{y}_{s\ell} v_{,xx} - \bar{z}_{s\ell} w_{,xx} + \tfrac{1}{2} w_{,x}^2 \right] dx \Big|_{\substack{y=y_\ell \\ z=\bar{z}}} \tag{50}$$

Putting the linear terms in the S vector and the nonlinear terms in the U matrix, as defined in Section 2.6, gives

$$\frac{\partial \Delta S}{\partial a_{1ij}} = -\sum_{\ell=1}^{L} P_{x\ell}\, Q_{1x21i}\, Y_{1j}(y_\ell) \tag{51}$$

$$\frac{\partial \Delta S}{\partial a_{2ij}} = +\sum_{\ell=1}^{L} P_{x\ell}\, \bar{a}^{1}\, \bar{y}_\ell\, Q_{1x32i}\, Y_{2j}(y_\ell) \tag{52}$$

$$\frac{\partial \Delta S}{\partial a_{3ij}} = +\sum_{\ell=1}^{L} P_{x\ell}\, \bar{a}^{1}\, \bar{z}_\ell\, Q_{1x33i}\, Y_{3j}(y_\ell) \tag{53}$$

$$\frac{\partial \Delta U}{\partial a_{1ij}} = \frac{\partial \Delta U}{\partial a_{2ij}} = 0 \tag{54}$$

$$\frac{\partial \Delta U}{\partial a_{3ij}} = -\sum_{\ell=1}^{L} \bar{a}^{1} P_{x\ell} \sum_m \sum_n \psi_{x23i3m} Y_{3j}(y_\ell) Y_{3n}(y_\ell)\, a_{3mn} \tag{55}$$

2.8.1.2 Rings

The energy terms for the discrete, eccentric rings are similar to those for the stringers, but are much more complicated. Reference [4] was also used for these energies; again, refer to Figures 3 and 4.

The potential energy is expressed as

$$\Delta V = \sum_{k=1}^{K} \frac{E_{rk}}{2} \int_0^b \Big[A_{rk} v_{,y}^2 + I_{xxrk} w_{,yy}^2 + I_{zzrk} u_{,yy}^2 + R^{-2} A_{rk} w^2 + R^{-2} I_{zzrk} w_{,x}^2 - 2\bar{z}_{rk} A_{rk} v_{,y} w_{,yy} - 2\bar{x}_{rk} A_{rk} \cdot$$

$$v_{,y} u_{,yy} + 2R^{-1} A_{rk} v_{,y} w + 2R^{-1} \bar{x}_{rk} A_{rk} v_{,y} w_{,x} + 2 I_{xzrk} w_{,yy} \cdot \tag{56}$$

$$u_{,yy} - 2R^{-1} \bar{z}_{rk} A_{rk} w_{,yy} w - 2R^{-1} I_{xzrk} w_{,yy} w_{,x} - 2R^{-1} \bar{x}_{rk} A_{rk} \cdot$$

$$u_{,yy} w - 2R^{-1} I_{zzrk} u_{,yy} w_{,x} + 2R^{-2} \bar{x}_{rk} A_{rk} w w_{,x} \Big]_{x=x_k} dy$$

$$+ \tfrac{1}{2} (GJ)_{rk} \int_0^b \left[w_{,xy}^2 \right]_{x=x_k} dy$$

The final forms of the partials required are

$$\frac{\partial \Delta V}{\partial a_{1ij}} = \sum_{k=1}^{K} \sum_m \sum_n E_{rk} \Big[b^{-3} I_{zzrk} X_{1i} X_{1m} \psi_{y31j1n} a_{1mn}$$

$$- b^{-2} \bar{x}_{rk} A_{rk} X_{1i} X_{2m} \psi_{y61j2n} a_{2mn} + \big(b^{-3} I_{xzrk} X_{1i} X_{3m} \psi_{y3j1n} \tag{57}$$

$$- \bar{x}_{rk} A_{rk} b^{-1} R^{-1} X_{1i} X_{3m} \psi_{y51j3n} - a^{-1} b^{-1} R^{-1} I_{zzrk} X_{1i} X_{3m,x} \psi_{y51j3n} \big) \cdot$$

$$a_{3mn} \Big]_{x=x_k}$$

$$\frac{\partial \Delta V}{\partial a_{2ij}} = \sum_{k=1}^{K} \sum_m \sum_n E_{rk} A_{rk} X_{2i} \Big[-b^{-2} \bar{x}_{rk} X_{1m} \psi_{y61n2j} a_{1mn}$$

$$+ b^{-1} X_{2m} \psi_{y22j2n} a_{2mn} + \big(-b^{-2} \bar{z}_{rk} X_{3m} \psi_{y63n2j} + R^{-1} X_{3m} \cdot \tag{58}$$

$$\psi_{y42j3n} + a^{-1} R^{-1} \bar{x}_{rk} X_{3m,x} \psi_{y42j3n} \big) a_{3mn} \Big]_{x=x_k}$$

$$
\begin{aligned}
\frac{\partial \Delta V}{\partial a_{3ij}} = \sum_{k=1}^{K} \sum_{m} \sum_{n} \Big\{ & E_{rk} \Big[\Big(b^{-3} I_{xzrk} X_{3i} X_{1m} \psi_{y33j1n} \\
& - b^{-1} R^{-1} \bar{x}_{rk} A_{rk} X_{3i} X_{1m} \psi_{y51n3j} - a^{-1} b^{-1} R^{-1} I_{zzrk} X_{3i,x} X_{1m} \cdot \\
& \psi_{y51n3j} \Big) a_{1mn} + \Big(-b^{-2} \bar{z}_{rk} A_{rk} X_{3i} X_{2m} \psi_{y63j2n} \\
& + R^{-1} A_{rk} X_{3i} X_{2m} \psi_{y42n3j} + a^{-1} R^{-1} \bar{x}_{rk} A_{rk} X_{3i,x} X_{2m} \psi_{y42n3j} \Big) \cdot \\
& a_{2mn} + \Big(b^{-3} I_{xxrk} X_{3i} X_{3m} \psi_{y33j3n} + b R^{-2} A_{rk} X_{3i} X_{3m} \cdot \\
& \psi_{y13j3n} + a^{-2} b R^{-2} I_{zzrk} X_{3i,x} X_{3m,x} \psi_{y13j3n} - b^{-1} R^{-1} \bar{z}_{rk} \cdot \\
& A_{rk} X_{3i} X_{3m} \psi_{y33j3n} - b^{-1} R^{-1} \bar{z}_{rk} A_{rk} X_{3i} X_{3m} \psi_{y53n3j} \\
& - a^{-1} b^{-1} R^{-1} I_{xzrk} X_{3i} X_{3m,x} \psi_{y63j3n} - a^{-1} b^{-1} R^{-1} I_{xzrk} X_{3i,x} X_{3m} \cdot \\
& \psi_{y53n3j} + a^{-1} b \bar{x}_{rk} R^{-2} A_{rk} X_{3i} X_{3m,x} \psi_{y13j3n} + a^{-1} b R^{-2} \bar{x}_{rk} \cdot \\
& A_{rk} X_{3i,x} X_{3m} \psi_{y13j3n} \Big) a_{3mn} \Big] + a^{-2} b (GJ)_{rk} \cdot \\
& \Big[X_{3i,x} X_{3m,x} \psi_{y23j3n} \Big] a_{3mn} \Big\}_{x=x_k}
\end{aligned}
\tag{59}
$$

The kinetic energy of the rings is expressed as

$$
\begin{aligned}
\Delta T = \tfrac{1}{2} \sum_{k=1}^{K} \rho_{rk} \int_0^b \Big[& A_{rk} \big(u_{,\tau}^2 - 2 \bar{z}_{rk} u_{,\tau} w_{,x\tau} + w_{,\tau}^2 \\
& + 2 \bar{x}_{rk} w_{,\tau} w_{,x\tau} + v_{,\tau}^2 - 2 \bar{z}_{rk} v_{,\tau} w_{,y\tau} - 2 \bar{x}_{rk} v_{,\tau} u_{,y\tau} \big) \\
& + I_{xxrk} \big(w_{,x\tau}^2 + w_{,y\tau}^2 \big) + 2 I_{xzrk} w_{,y\tau} u_{,y\tau} \\
& + I_{zzrk} \big(w_{,x\tau}^2 + u_{,y\tau}^2 \big) \Big]_{x=x_k} dy
\end{aligned}
\tag{60}
$$

The final forms of the ring kinetic energy partial derivatives are

$$
\begin{aligned}
\frac{\partial \Delta T}{\partial a_{1ij}} = \sum_{k=1}^{K} \sum_{m} \sum_{n} \rho_{rk} \omega^2 \Big\{ & \left[b A_{rk} \psi_{y11j1n} + b^{-1} I_{zzrk} \psi_{y21j1n} \right] \cdot \\
& X_{1i} X_{1m} a_{1mn} - \bar{x}_{rk} A_{rk} X_{1i} X_{2m} \psi_{y41j2n} a_{2mn} + \left[-\bar{z}_{rk} a^{-1} b A_{rk} \cdot \right. \\
& \left. X_{3m,x} \psi_{y11j3n} + b^{-1} I_{xzrk} X_{3m} \psi_{y21j3n} \right] X_{1i} a_{3mn} \Big\}_{x=x_k}
\end{aligned} \tag{61}
$$

$$
\begin{aligned}
\frac{\partial \Delta T}{\partial a_{2ij}} = \sum_{k=1}^{K} \sum_{m} \sum_{n} \rho_{rk} \omega^2 A_{rk} \Big\{ & \left[-\bar{x}_{rk} X_{2i} X_{1m} \psi_{y41n2j} \right] a_{1mn} \\
& + b X_{2i} X_{2m} \psi_{y12j2n} a_{2mn} - \bar{z}_{rk} X_{2i} X_{3m} \psi_{y43n2j} a_{3mn} \Big\}_{x=x_k}
\end{aligned} \tag{62}
$$

$$
\begin{aligned}
\frac{\partial \Delta T}{\partial a_{3ij}} = \sum_{k=1}^{K} \sum_{m} \sum_{n} \rho_{rk} \omega^2 \Big\{ & \left[-a^{-1} b \bar{z}_{rk} A_{rk} X_{3i,x} \psi_{y13j1n} \right. \\
& \left. + b^{-1} I_{xzrk} X_{3i} \psi_{y23j1n} \right] X_{1m} a_{1mn} - \left[\bar{z}_{rk} A_{rk} X_{3i} X_{2m} \cdot \right. \\
& \left. \psi_{y43j2n} \right] a_{2mn} + \Big[(\psi_{y13j3n}) \big(b A_{rk} X_{3i} X_{3m} \\
& + a^{-1} b \bar{x}_{rk} A_{rk} \left[X_{3i} X_{3m,x} + X_{3i,x} X_{3m} \right] + a^{-2} b X_{3i,x} \cdot \\
& X_{3m,x} \left[I_{xxrk} + I_{zzrk} \right] \big) + b^{-1} I_{xxrk} X_{3i} X_{3m} \psi_{y23j3n} \Big] \cdot \\
& a_{3mn} \Big\}_{x=x_k}
\end{aligned} \tag{63}
$$

For a panel rather than a complete cylinder, a ring stiffener may support a circumferential load, P_y, imposed at its ends. The energy associated with P_y, which is positive in tension, is given by

$$
\begin{aligned}
\Delta U = - b P_y \int_y & \left[b^{-1} (v_{,y} - z w_{,yy} - \bar{x} u_{,yy}) + (R^{-1} w + R^{-1} \bar{x} w_{,x}) \right. \\
& \left. + \tfrac{1}{2} b^{-2} w_{,y}^2 \right] d\!\left(\frac{y}{b}\right) \Bigg|_{\substack{x=\bar{x}_k \\ z=\bar{z}}}
\end{aligned} \tag{64}
$$

After separating into linear and nonlinear terms, as for the stringers, the final partial derivatives are given as

$$\frac{\partial \Delta S}{\partial a_{1ij}} = -\sum_{k=1}^{K} P_{yk}\, \bar{x}_k\, X_{1i}(x_k)\, \Phi_{1y31j} \tag{65}$$

$$\frac{\partial \Delta S}{\partial a_{2ij}} = -\sum_{k=1}^{K} P_{yk}\, X_{2i}(x_k)\, \Phi_{y22j} \tag{66}$$

$$\frac{\partial \Delta S}{\partial a_{3ij}} = -\sum_{k=1}^{K} P_{yk} \Big[-\bar{z}_k X_{3i}(x_k)\, \Phi_{y33j} + bR^{-1} X_{3i}(x_k)\, \Phi_{1y13j} + bR^{-1} \bar{x}_k X_{3i,x}(x_k)\, \Phi_{1y13j} \Big] \tag{67}$$

$$\frac{\partial \Delta U}{\partial a_{1ij}} = \frac{\partial \Delta U}{\partial a_{2ij}} = 0 \tag{68}$$

$$\frac{\partial \Delta U}{\partial a_{3ij}} = -\sum_{k=1}^{K} \sum_{m} \sum_{n} P_{yk}\, b^{-1} X_{3i}(x_k) X_{3m}(x_k)\, \Psi_{y33j3n}\, a_{3mn} \tag{69}$$

2.8.2 Lumped Masses

The kinetic energy contribution of each lumped mass attached to the shell is written in terms of its translational inertia only as

$$\Delta T = \tfrac{1}{2}\, \bar{m} \left(u_{,\tau}^2 + v_{,\tau}^2 + w_{,\tau}^2 \right) \tag{70}$$

In final partial form, after using the assumed mode definitions of u, v, and w,

$$\frac{\partial \Delta T}{\partial a_{1ij}} = \bar{m}\, \omega^2 X_{1i} Y_{1j} \sum_{m} \sum_{n} X_{1m} Y_{1n} a_{1mn} \Big|_{pt.} \tag{71}$$

$$\frac{\partial \Delta T}{\partial a_{2ij}} = \bar{m}\omega^2 X_{2i} Y_{2j} \sum_m \sum_n X_{2m} Y_{2n} a_{2mn} \Big|_{@\,pt.} \tag{72}$$

$$\frac{\partial \Delta T}{\partial a_{3ij}} = \bar{m}\omega^2 X_{3i} Y_{3j} \sum_m \sum_n X_{3m} Y_{3n} a_{3mn} \Big|_{@\,pt.} \tag{73}$$

2.8.3 Spring Supports

To model nonstandard boundary or internal attachment conditions, it is convenient to have the capability to introduce discrete point and line spring supports.

2.8.3.1 At a Point

Assuming that the spring acts normal to the shell surface, its energy can be defined in terms of w only as

$$\Delta V = \tfrac{1}{2} K_p w^2 \Big|_{@\,pt.} \tag{74}$$

The partial derivatives are then trivially formed as

$$\frac{\partial \Delta V}{\partial a_{1ij}} = \frac{\partial \Delta V}{\partial a_{2ij}} = 0 \tag{75}$$

$$\frac{\partial \Delta V}{\partial a_{3ij}} = K_p X_{3i} Y_{3j} \sum_m \sum_n X_{3m} Y_{3n} a_{3mn} \Big|_{@\,pt.} \tag{76}$$

2.8.3.2 Along a Line

To simplify, it is assumed that the line spring supports lie parallel to either the x- or y-axis of the shell. Then,

$$\Delta V = \begin{cases} \tfrac{1}{2} K_L\, a \int_0^1 w^2\, d\left(\frac{x}{a}\right) & ; \text{ x-axis} \\ \tfrac{1}{2} K_L\, b \int_0^1 w^2\, d\left(\frac{y}{b}\right) & ; \text{ y-axis} \end{cases} \tag{77}$$

After integration and partial differentiation,

$$\frac{\partial \Delta V}{\partial a_{1ij}} = \frac{\partial \Delta V}{\partial a_{2ij}} = 0 \tag{78}$$

$$\frac{\partial \Delta V}{\partial a_{3ij}} = \begin{cases} \sum_m \sum_n K_L a \psi_{x13i3m} Y_{3j} Y_{3n} a_{3mn} \Big|_{y=y_{L.S.}} & ;\text{x-axis} \\ \sum_m \sum_n K_L b X_{3i} X_{3m} \psi_{y13j3n} a_{3mn} \Big|_{x=x_{L.S.}} & ;\text{y-axis} \end{cases} \tag{79}$$

2.8.4 Concentrated Loads

The potential energy of a point load applied normal to the shell surface is written simply as

$$\Delta Q = P_c w \Big|_{@\,pt.} \tag{80}$$

The final partial form is just as simply written as

$$\frac{\partial \Delta Q}{\partial a_{1ij}} = \frac{\partial \Delta Q}{\partial a_{2ij}} = 0 \tag{81}$$

$$\frac{\partial \Delta Q}{\partial a_{3ij}} = P_c X_{3i} Y_{3j} \Big|_{@\,pt.} \tag{82}$$

2.8.5 Concentrated Moments

The energy associated with concentrated moment loading is important when input loading from attached members must be assessed. In both point and line moment cases, the vector describing the direction of the moment must be parallel to either the x or y-axis of the shell.

2.8.5.1 <u>At a Point</u>

The energy is formed as the product of the applied moment and the angle through which it is applied

$$\Delta Q = \begin{cases} +M_p\left(\bar{a}^{-1} w_{,x} + \bar{R}^{-1} u\right) & ; \vec{M}_p \text{ IN Y-DIR.} \\ -M_p\left(\bar{b}^{-1} w_{,y} + \bar{R}^{-1} v\right) & ; \vec{M}_p \text{ IN X-DIR.} \end{cases} \tag{83}$$

Substitution of the displacement definitions and partial differentiation of the energy gives

$$\frac{\partial \Delta Q}{\partial a_{1ij}} = \begin{cases} M_p \bar{R}^{-1} X_{1i} Y_{1j}\Big|_{@pt.} & ; \vec{M}_p \text{ IN Y-DIR.} \\ 0 & ; \vec{M}_p \text{ IN X-DIR.} \end{cases} \tag{84}$$

$$\frac{\partial \Delta Q}{\partial a_{2ij}} = \begin{cases} 0 & ; \vec{M}_p \text{ IN Y-DIR.} \\ -M_p \bar{R}^{-1} X_{2i} Y_{2j}\Big|_{@pt.} & ; \vec{M}_p \text{ IN X-DIR.} \end{cases} \tag{85}$$

$$\frac{\partial \Delta Q}{\partial a_{3ij}} = \begin{cases} \bar{a}^{-1} M_p X_{3i,x} Y_{3j}\Big|_{@pt.} & ; \vec{M}_p \text{ IN Y-DIR.} \\ -\bar{b}^{-1} M_p X_{3i} Y_{3j,y}\Big|_{@pt.} & ; \vec{M}_p \text{ IN X-DIR.} \end{cases} \tag{86}$$

2.8.5.2 <u>Along a Line</u>

The formation of the energy contribution for line moments is the same as that for point moments except that a line integral is required. The final partials are

$$\frac{\partial \Delta Q}{\partial a_{1ij}} = \begin{cases} M_L b \bar{R}^{-1} X_{1i} \Phi_{y_{1j}}\Big|_{x=x_{LM}} & ; \vec{M}_L \text{ IN Y-DIR.} \\ 0 \end{cases} \tag{87}$$

$$\frac{\partial \Delta Q}{\partial a_{2ij}} = \begin{cases} 0 \\ -M_L a \bar{R}^{-1} \Phi_{x_{2i}} Y_{2j}\Big|_{y=y_{LM}} & ; \vec{M}_L \text{ IN X-DIR.} \end{cases} \tag{88}$$

$$\frac{\partial \Delta Q}{\partial a_{3ij}} = \begin{cases} M_L \bar{a}^1 b\, X_{3i,x}\, \Phi_{1y,3j}\big|_{x=x_{Lm}} & ;\ \vec{M}_L \text{ IN Y-DIR.} \\ -M_L a \bar{b}^1 \Phi_{1x,3i}\, Y_{3j,Y}\big|_{y=y_{Lm}} & ;\ \vec{M}_L \text{ IN X-DIR.} \end{cases} \tag{89}$$

2.9 BOUNDARY CONDITIONS

The boundary conditions to be considered are the classical conditions of clamped, simply supported, or free. All combinations of these three may be specified, that is, any edge of a panel may be specified as clamped, supported, or free. In addition, any two opposite edges may have elastic moment restraint. A distinct advantage of the Rayleigh-Ritz method is that only the geometric boundary conditions (displacement and slope) need be satisfied to insure convergence of the solution (although convergence is improved by the satisfaction of the force boundary conditions). The Rayleigh-Ritz method does require a set of assumed modal functions, each of which satisfies the geometric boundary conditions. The functions chosen for this study are a series of simple beam vibration modes. These functions form a complete orthogonal set, and are all of the same general form. The use of these functions allows the normal deflection, w, to satisfy the following conditions:

(1) clamped edge: $w = 0;\ w, n = 0$

(2) simply supported edge: $w = 0;\ w,_{nn} = 0$

(3) free edge: $w,_{nn} = 0;\ w,_{nnn} = 0$

(4) elastically restrained edge: $w = 0;\ w,_{nn} = \alpha w,_{n}$

where n denotes a normal to the particular edge.

In addition to these conditions, which apply to flat or curved plates and the ends of a cylinder, the normal deflection in the circumferential direction of a cylinder is taken to be

$$Y_{3n} = \cos \frac{2n\pi Y}{b} \tag{90}$$

An assumption has been made concerning the form of u and v. In the x direction, it is assumed that the mode shape function for v is the same as that for w and that the mode shape function for u is the derivative of that for w. Mathematically,

$$X_{1m} = X_{3m,x}$$
$$X_{2m} = X_{3m} \tag{91}$$

Since the roles of u and v are reversed in the y direction, it is also assumed that

$$Y_{1n} = Y_{3n}$$
$$Y_{2n} = Y_{3n,y} \tag{92}$$

These assumptions on the form of u and v allow them to always satisfy their required geometric boundary conditions. The specific form of the assumed modes and the evaluation of the necessary integrals is discussed further in Section 2.10.

In connection with the free-edge boundary condition, it must be noted that no geometric boundary conditions (deflection or slope) on the w displacement are specified. In addition, the force boundary conditions for the free edge of an anisotropic curved panel are so complicated as to be impossible to satisfy with modal functions as simple as beam modes. Thus, while the use of the beam mode functions for a free edge is intuitively acceptable, some difficulties are to be expected.

Often the boundary restraint provided by real structure is between the classical simple support and clamped conditions. Particularly in vibration problems, modeling the actual edge restraint can be important.

The inclusion of elastic moment restraint follows the approach used by Ashton in References [1], [5], and [6]. Basically, a beam mode function having the appropriate frequency and mode shape for the input elastic restraint parameters is calculated (Reference [1]). In addition, the potential energy absorbed by the boundaries must be combined with the usual strain energy.

If the edge restraint moment (in the x-direction) is assumed to be of the form

$$M_x \approx \alpha_x D_{11} w_{,x} \tag{93}$$

Then the potential energy contribution is of the form

$$\Delta V = \frac{1}{2} \int M_x \, w_{,x} \tag{94}$$

$$\Delta V = \frac{1}{2} \alpha_x D_{11} \int w_{,x}^2 \, dx \tag{95}$$

Generalization of this form at both x-edges and both y-edges gives rise to the following approximate potential energy increment:

$$\begin{aligned} \Delta V = & \tfrac{1}{2} D_{11} \left[\alpha_x b \int_0^1 a^{-3} w_{,x}^2 \, d\left(\tfrac{y}{b}\right) \Big|_{x=0} + \beta_x b \int_0^1 a^{-3} w_{,x}^2 \, d\left(\tfrac{y}{b}\right) \Big|_{x=a} \right] \\ & + \tfrac{1}{2} D_{22} \left[\alpha_y a b^{-3} \int_0^1 w_{,y}^2 \, d\left(\tfrac{x}{a}\right) \Big|_{y=0} + \beta_y a b^{-3} \int_0^1 w_{,y}^2 \, d\left(\tfrac{x}{a}\right) \Big|_{y=b} \right] \end{aligned} \tag{96}$$

where

$$\begin{aligned} \alpha_x &= K_{x1} a / D_{11} \\ \beta_x &= K_{x2} a / D_{11} \\ \alpha_y &= K_{y1} b / D_{22} \\ \beta_y &= K_{y2} b / D_{22} \end{aligned} \tag{97}$$

and K_{x1}, K_{x2}, K_{y1}, K_{y2} are rotational spring constants (in-lb./rad/in) which characterize the support stiffness. The final form of the varied potential energy is

$$\frac{\partial \Delta V}{\partial a_{1ij}} = \frac{\partial \Delta V}{\partial a_{2ij}} = 0 \tag{98}$$

$$\begin{aligned} \frac{\partial \Delta V}{\partial a_{3ij}} = \sum_m \sum_n \Big\{ & a^{-3} b D_{11} \Psi_{y13j3n} \left[\alpha_x X_{3i,x} X_{3m,x} \big|_{x=0} \right. \\ & \left. + \beta_x X_{3i,x} X_{3m,x} \big|_{x=a} \right] + a b^{-3} D_{22} \Psi_{x13i3m} \cdot \\ & \left[\alpha_y Y_{3j,y} Y_{3n,y} \big|_{y=0} + \beta_y Y_{3j,y} Y_{3n,y} \big|_{y=b} \right] \Big\} a_{3mn} \end{aligned} \tag{99}$$

2.10 EVALUATION OF INTEGRALS

As shown in Reference [1], the beam mode shapes can be written as a sum of four terms as follows:

$$Z_m(z) = \sum_{j=1}^{4} C_{mj}\,\rho_{jm} \tag{100}$$

where

$$\begin{aligned} \rho_{1m} &= \mathrm{COSH}\,(\epsilon_m z) \\ \rho_{2m} &= \mathrm{COS}\,(\epsilon_m z) \\ \rho_{3m} &= \mathrm{SINH}\,(\epsilon_m z) \\ \rho_{4m} &= \mathrm{SIN}\,(\epsilon_m z) \end{aligned} \tag{101}$$

and the C_{mj} are constants for the particular mode shape m and the appropriate boundary condition. The ϵ_m is the corresponding natural frequency of the m^{th} mode. The C_{mj} constants are tabulated in Reference [1]. The successive derivatives of $Z_m(z)$ are also of this form with changes in the C_{mj} due to the repeating nature of the derivatives of the ρ_{jm}. The z-notation used here is replaced by x or y depending on the plate direction being integrated.

With this special form of the beam mode shapes all of the various integrals may be obtained in a closed form. The detailed solution method is documented in Reference [1].

Since the u and v displacement functions are assumed to be of the same form as w or its derivatives, all of the functions used can be integrated by the same solution technique.

The definition of the integral terms used throughout the analysis to denote the product of two functions is

$$\psi_{zkijmn} \equiv \int_0^1 Z_{ij,\ell z}\, Z_{mn,pz}\, dz \tag{102}$$

where the k subscript defines the number of derivatives as shown in Table I.

Table I. DEFINITIONS OF ℓ AND p VERSUS k

GIVEN	DEFINES	
k	ℓ	p
1	0	0
2	1	1
3	2	2
4	1	0
5	2	0
6	2	1

For example,

$$\psi_{x41j3n} = \int_0^1 X_{1j,x} X_{3m} \, dx . \tag{103}$$

The notation used to denote those integrals in which two w-functions are integrated in the presence of a power term (Section 2.6) is

$$\Omega_{kijmn} \equiv \int_0^1 z^{k-1} Z_{3m,\ell z} Z_{3n,pz} \, dz \tag{104}$$

where i = 1 means that z stands for x and i = 2 means that z stands for y. The relationship between j on the left side and ℓ and p on the right side is given by Table II.

Table II. DEFINITIONS OF ℓ AND p VERSUS j

GIVEN	DEFINED	
j	ℓ	p
1	0	0
2	1	1
3	1	0

The notation used to denote a single mode integrated in the presence of a power is given by

$$\Phi_{izrgm} \equiv \int_0^1 z^{i-1} Z_{gm,(r-1)z}\, dz \tag{105}$$

The only integral evaluations involving deviations from the solution format are the rigid body modes necessary for the simple-free and free-free boundary conditions.

For the simple-free case in the x-direction

$$X_{11} = \sqrt{3}$$
$$X_{21} = \sqrt{3}\, x \tag{106}$$
$$X_{31} = \sqrt{3}\, x$$

or in the y direction,

$$Y_{11} = \sqrt{3}\, y$$
$$Y_{21} = \sqrt{3} \tag{107}$$
$$Y_{31} = \sqrt{3}\, y$$

These mode functions must be combined with the standard form mode functions in a special integral table.

For the free-free boundary condition in the x-direction,

$$X_{11} = 0 \qquad X_{12} = -2\sqrt{3}$$
$$X_{21} = 1 \qquad X_{22} = \sqrt{3}(1-2x) \tag{108}$$
$$X_{31} = 1 \qquad X_{32} = \sqrt{3}(1-2x)$$

or in the y-direction,

$$Y_{11} = 1 \qquad Y_{12} = \sqrt{3}(1-2y)$$
$$Y_{21} = 0 \qquad Y_{22} = -2\sqrt{3} \tag{109}$$
$$Y_{31} = 1 \qquad Y_{32} = \sqrt{3}(1-2y)$$

As in the simple-free case, these mode functions must be combined with the standard form mode function in a special integral table.

SECTION III

ANALYTICAL AND EXPERIMENTAL CORRELATION

The results of many problem solutions using Procedure SS8 are described in this section. During the development and checkout stages, runs were made to simulate rectangular beams and flat plates. These runs served to debug minor programming errors and build confidence in the solution techniques employed. Subsequent runs were made to compare with existing theoretical and experimental results for isotropic shell segments and cylinders. To test the laminated anisotropic capabilities of the program, it was necessary to perform an experimental test program and to borrow results from on-going composites programs. These tests brought program limitations to light, some of which were overcome and some of which remain.

3.1 STATIC DEFLECTION

The static deflection of an anisotropic plate was checked against Procedure RA5, now revised to be Procedure SOO. Agreement was good in all cases. No experimental results for shells could be found, so an experimental program was undertaken.

Two types of tests were performed to assess static deflection capabilities. In performing the first set, done under the Fuselage Program, deflection of fully clamped curved panels under a uniform pressure load was sought. In performing the second set, done under the Dynamic Characteristics Program, Contract F33615-70-C-1837, the determination of the influence coefficients of cantilever curved panels was sought.

3.1.1 Fuselage Program Tests

All of the advanced composite curved plate specimens were laminated graphite-epoxy and fabricated according to drawing number FW6915067. All specimens had the same geometric configuration with respect to length, width and curvature; a sketch of a typical specimen is shown in Figure 5. Average thicknesses and laminate designs of the panels were the physical variables for this program.

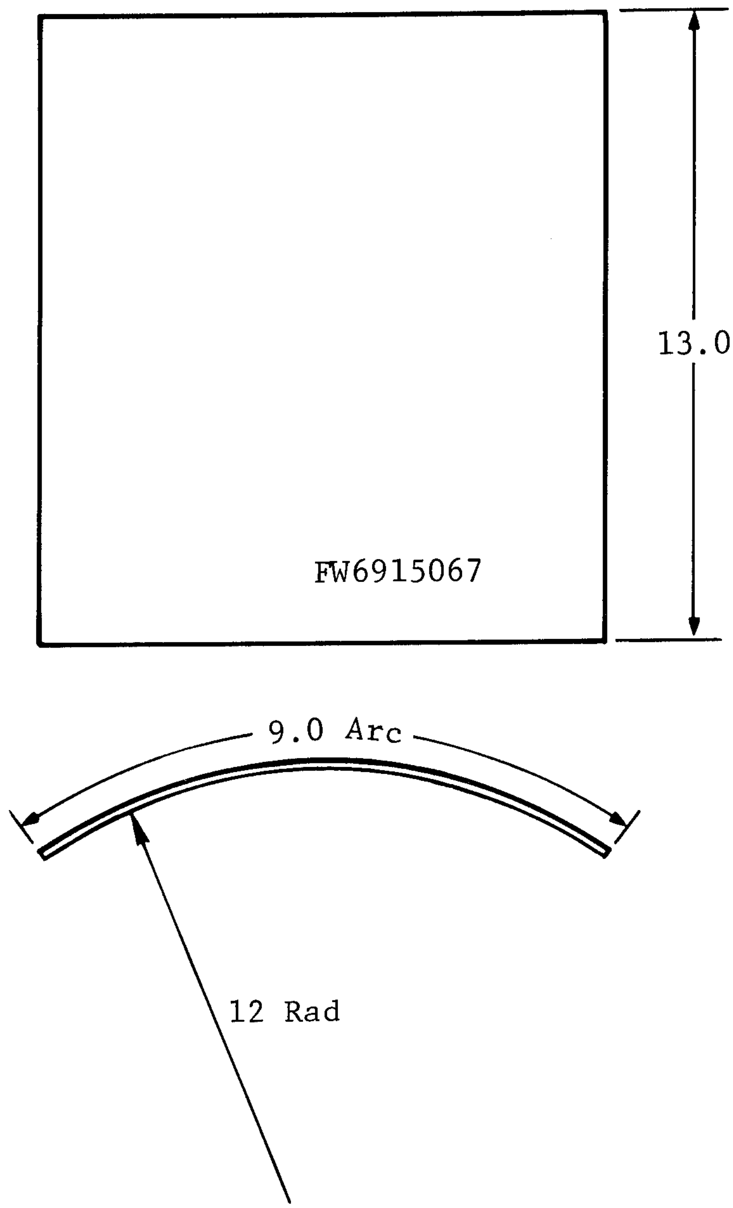


Figure 5 Fuselage Program Curved Panel Specimen Geometry

The specimens were hand-laid using Morganite II/4617 which has lamina properties of

$$E_1 = 20 \times 10^6 \text{ psi}$$

$$E_2 = 2.1 \times 10^6 \text{ psi}$$

$$G_{12} = 0.85 \times 10^6 \text{ psi}$$

$$\nu_{12} = 0.21$$

After their layup on a table, multiple-specimens were draped into a concave steel tool, bagged, and cured. The final operations were to net-trim the straight edges on a specially jigged table saw and net trim the curved edges with an end-mill.

The test fixture was not only used for the lateral pressure tests but was also used for compression buckling and vibration tests. It provided clamped-clamped boundary conditions for the curved edges and either clamped-clamped or simple-simple conditions for the straight edges. Clamping bars provided for variations in thickness of the panels. The test fixture is shown in Figures 6 through 9.

The set-up for the lateral load, or pressure, test utilized a rubber pressure bag mounted against the concave side of the panel. The back side of the bag was reacted with a stiffened pressure plate having the same contour as the panel and bolted to the fixture's side support (see Figures 10 through 13). The size of the bag, when deflated, was sufficient to cover the unsupported area of the panel without creasing or stretching, and thus provided an even load distribution over the face of the panel as air pressure was increased. During the pressure application, the load machine maintained a 100-pound edge load. After preliminary runs using a dial gage, an LVDT instrument measured the out-of-plane deflections as the pressure was increased. Measurements were recorded at increasing pressure increments. For these tests, the panel edges were fully clamped.

The test results and analytical predictions are shown in Table III in terms of center-deflection-per-psi of pressure. The load deflection plots are detailed in Reference [7]. Some analytical results with simply supported straight edges are included in Table III to indicate the sensitivity to boundary conditions. Correlation with elastically restrained edges was not attempted. It is obvious by the poor correlation that the fully clamped boundary condition was not properly modeled in the tests.

SMD7040

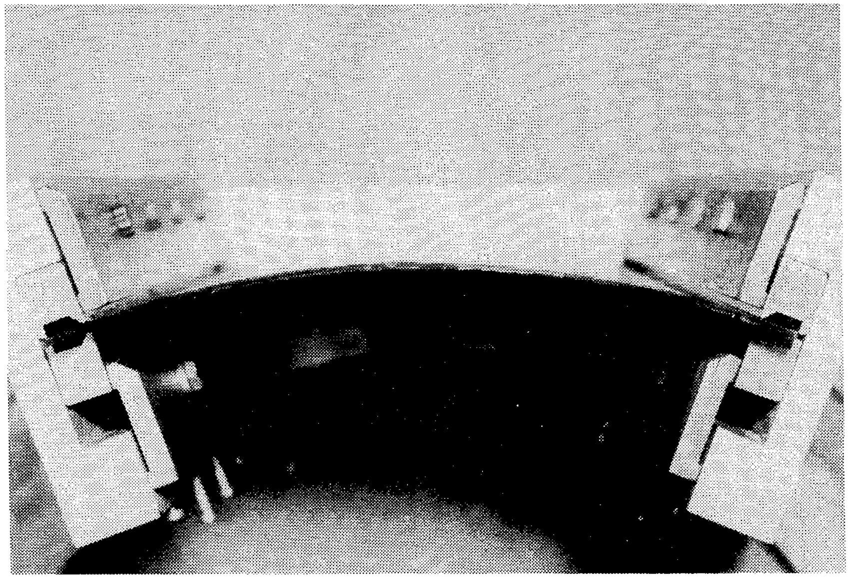


Figure 6 Top View of Test Fixture Showing Simply-Supported Sides

SMD7041

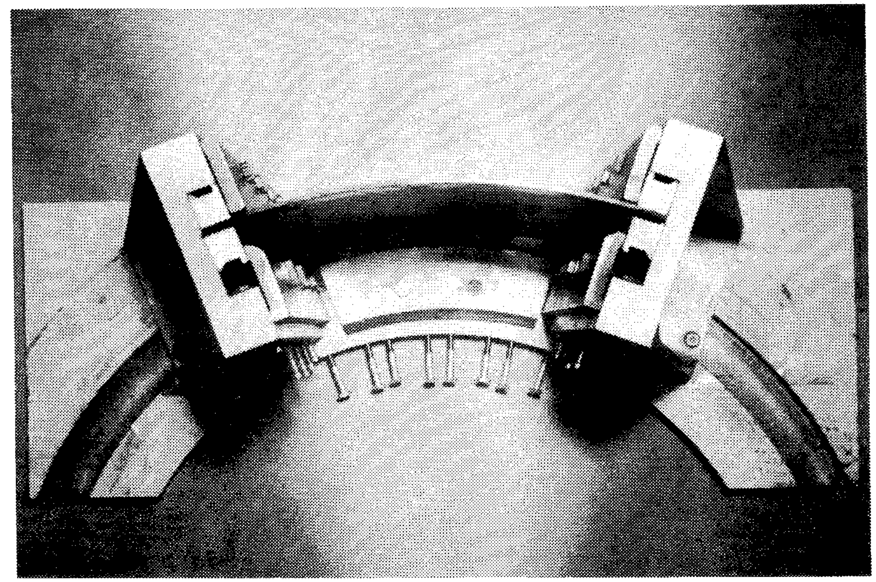


Figure 7 Top View of Test Fixture Showing Clamped Sides

SMD7042

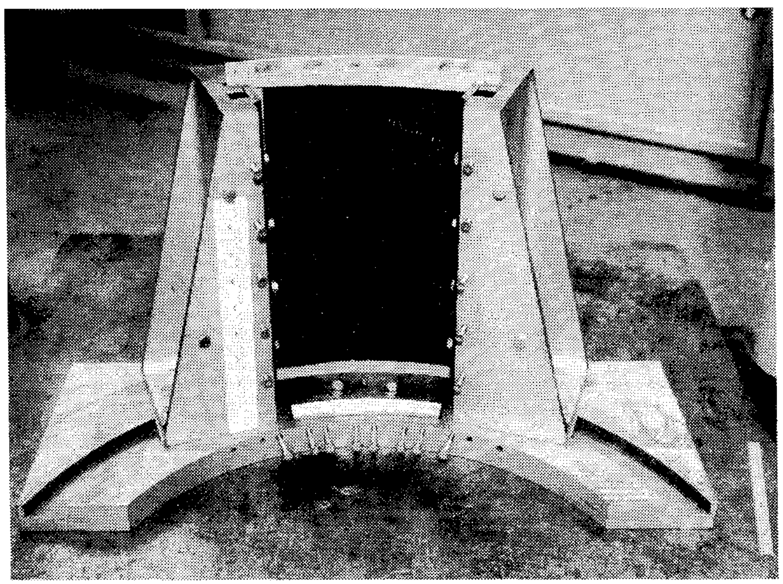


Figure 8 Front View of Test Fixture

SMD7043

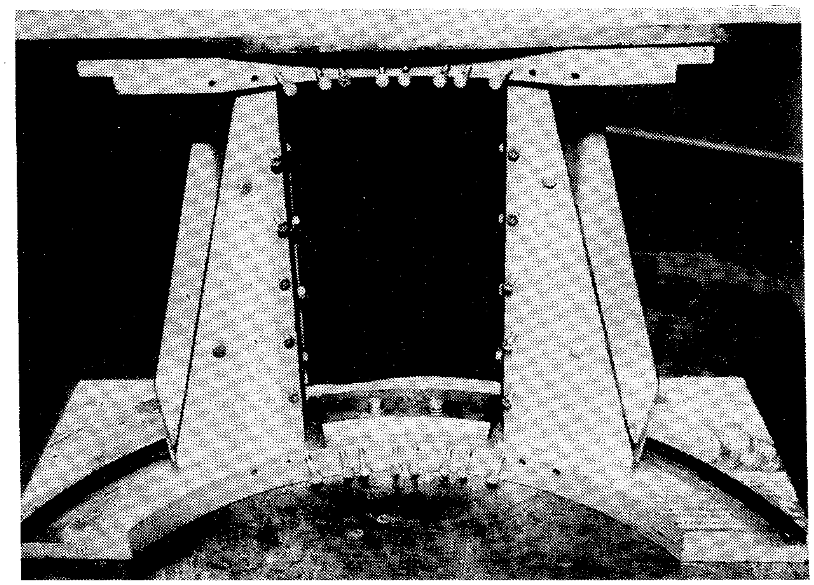


Figure 9 Front View of Test Fixture with Top Support Installed

SMD7044

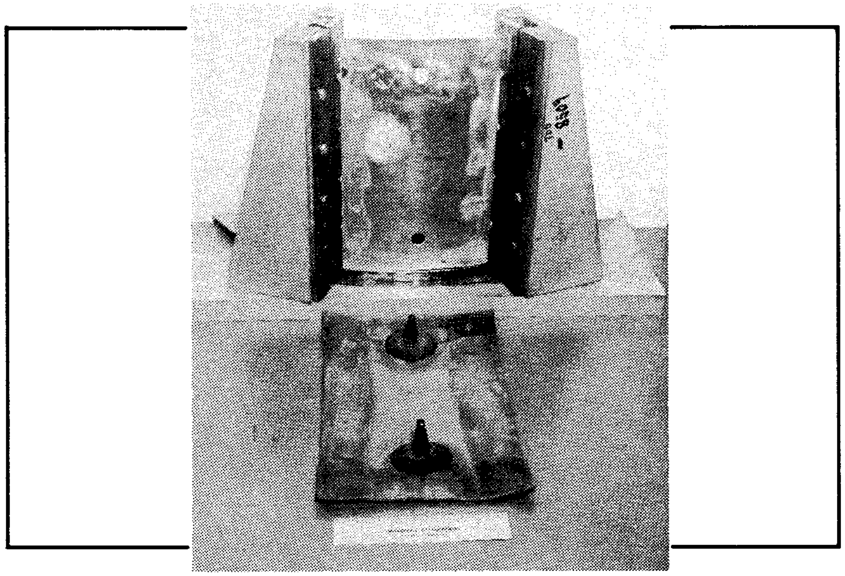


Figure 10 Backup Structure and Pressure Bag

SMD7045

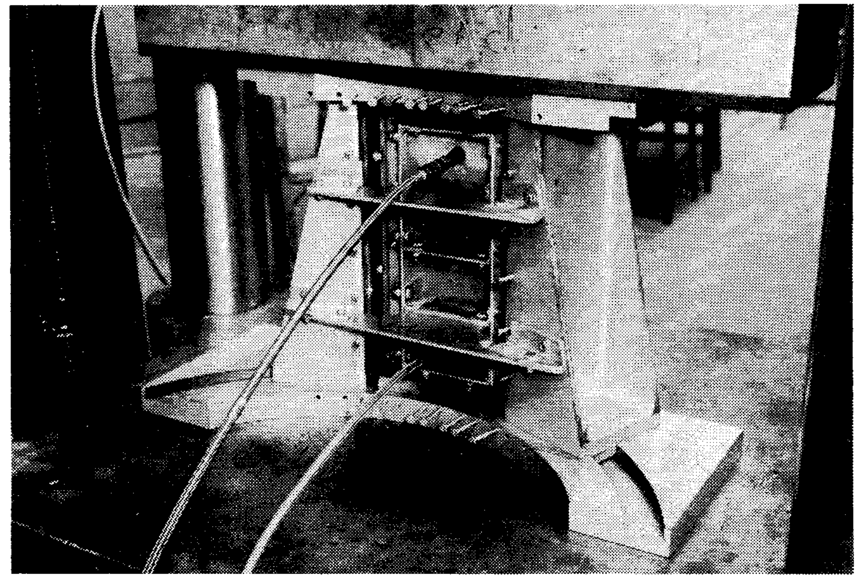


Figure 11 Assembled Pressure Fixture

SMD7046

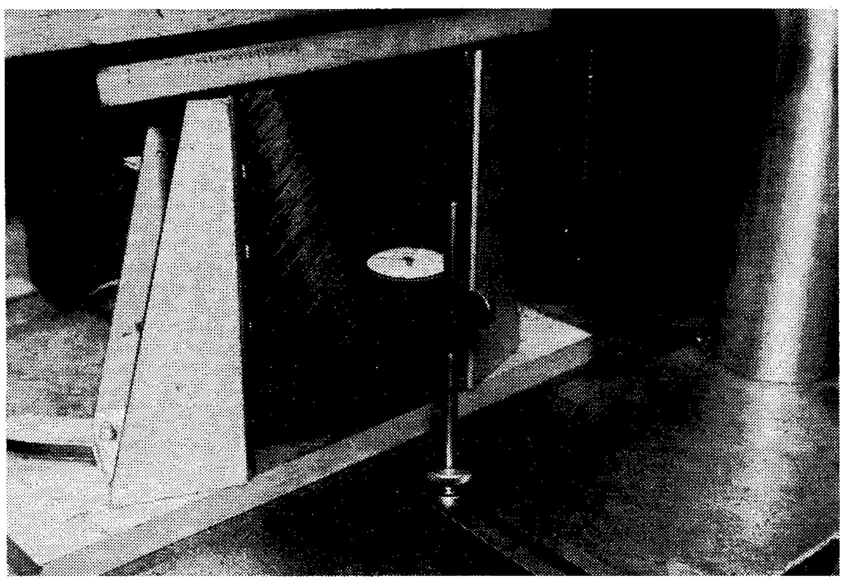


Figure 12 Deflection Measurement Setup

SMD7047

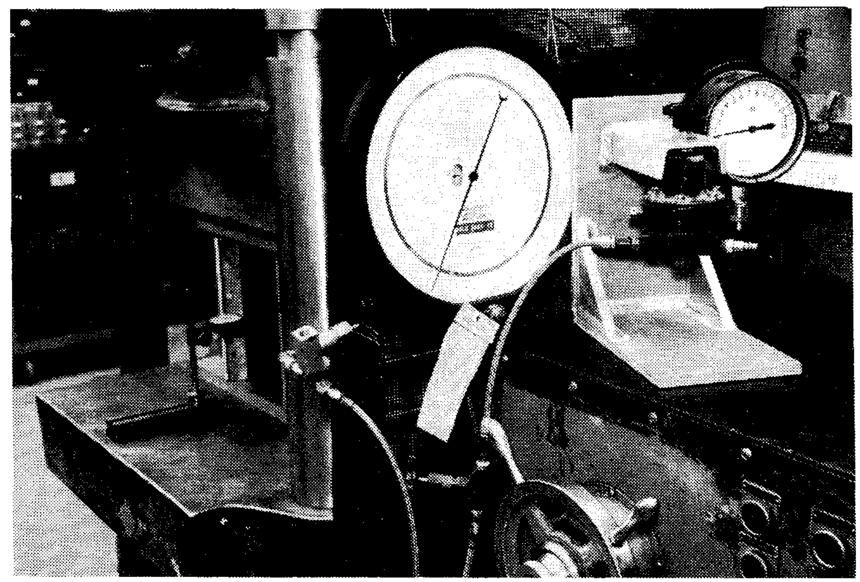


Figure 13 General View of Pressure Test Equipment

Table III PRESSURE TEST RESULTS

PANEL	LAMINATE	t	W/q mils/psi				
			EXP.	CCCC		CCSS	
				MAX.	CENTER	MAX.	CENTER
19A	$[\pm 45]_{2s}$	.0696	2.23	0.54	0.54		
19D	$[\pm 45]_{2s}$	.0719	2.25	0.52	0.52		
21A	$[0,90]_{s}$	.0289	9.20	0.83	0.83	36.2	5.6
23E	$[\pm 45]_{s}$	.0307	5.01	1.25	0.89		
29E	$[\pm 45]_{3s}$	.0892	2.63	0.43	0.43		
33E	$[\pm 45]_{4s}$	.0591	4.8	0.67	0.67	4.3	1.6
35A	$[-45]_{12}$	.0902	2.87	0.45	0.45	2.1	1.5
39A	$[+30]_{8}$	.0580	4.70	1.35	1.29	5.2	0.44
41A	$[+30]_{12}$	.0900	3.03	0.91	0.91	2.3	1.1
45E	$[0]_{8}$	.0582	6.67	2.46	2.46	13.7	7.3
49A	$[0,90]_{3s}$	.0880	3.88	0.28	0.28	6.7	6.7
51A	$[\pm 30]_{s}$	.0296	7.52	2.76	1.67	19.8	-7.2
53A	$[\pm 30]_{2s}$	.0557	3.16	1.27	1.23	5.1	-.062
55A	$[\pm 30]_{4s}$	.0807	2.40	0.97	0.97	2.4	.78
59A	$[0,\pm 60]_{s}$	.0422	3.43	0.62	0.62	8.7	-1.02

3.1.2 Dynamic Characteristics Program Tests

Three curved cantilever panel specimens were designed to study the effect of curvature, in the presence of material anisotropy, on the response of composite structures. The specimens are designated 15, 16A, and 16B and are shown in Figures 14 through 16. All of the curved specimens have 15-inch spans and 24-ply, $[0/\pm45_4/90]$ laminates. Specimen 15 has a 15-inch chord and a 36-inch radius, while Specimens 16A and 16B have 6-inch chords and 36- and 12-inch radii, respectively. A detailed explanation of these tests is given in Reference [13].

Some difficulty was experienced in conducting influence coefficient testing for the curved panels. A special fixture was developed with which the point loads normal to the undeflected middle surface of the specimen, i.e., in the radial direction, could be applied. Since vertical, free-floating Linear Variable Differential Transformers (LVDT's) were used for deflection measurements, the deflections were not measured radially. The LVDT's were inclined to the vertical as much as possible without compromising the accuracy of the instruments, but they could not be used in the radial direction. The maximum error in deflection caused by this setup was approximately two percent along each edge of the specimen.

Geometric nonlinearities caused by large chordwise cambering deflections were observed in these specimens, particularly in Specimen 15. This was indicated by the lack of symmetry in the off-diagonal terms of the influence coefficients as presented in Table IV. The notation DRR signifies Direct Rayleigh-Ritz, which is SS8. The notation USA denotes Unified Structural Analysis, a finite element procedure. It can be seen in the table that SS8 models the bending stiffness better than the finite element procedure, but does worse for the torsional stiffness. Generally, the correlation was rather poor, but no cause for this could be found. The problem is suspected to be that for these panels the chordwise boundary conditions are free-free, and the free-free modes are not operating properly.

3.2 STABILITY

The stability option of Procedure SS8 is the most important option available to the composites analyst and designer. The fact that composite shells exhibit complicated coupling between material and geometric stiffness effects precludes the use of simple design formulas.

SMD7048

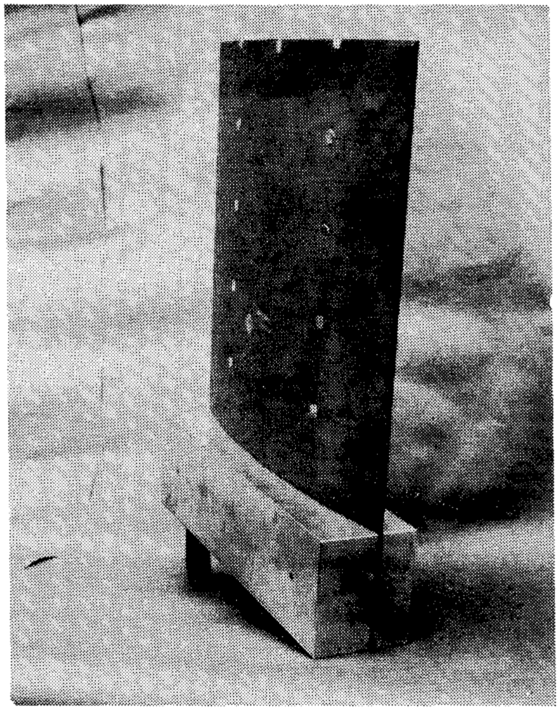


Figure 14 Curved Panel - Specimen 15

SMD7049

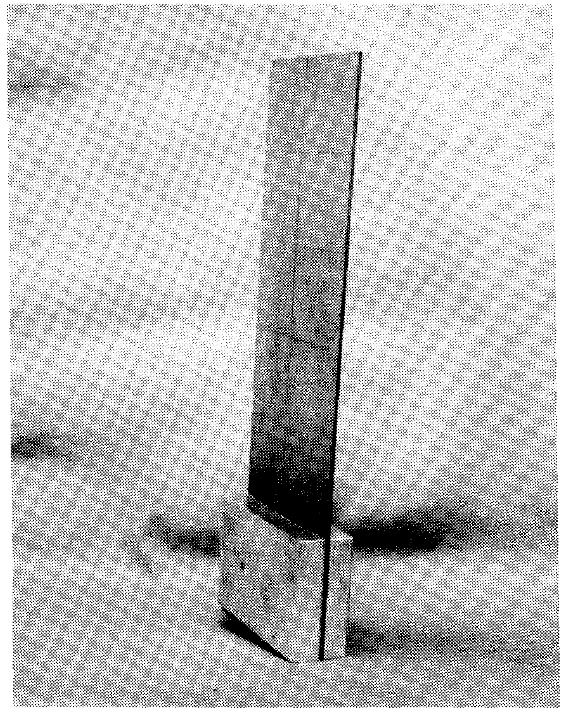


Figure 15 Curved Panel - Specimen 16A

SMD7050

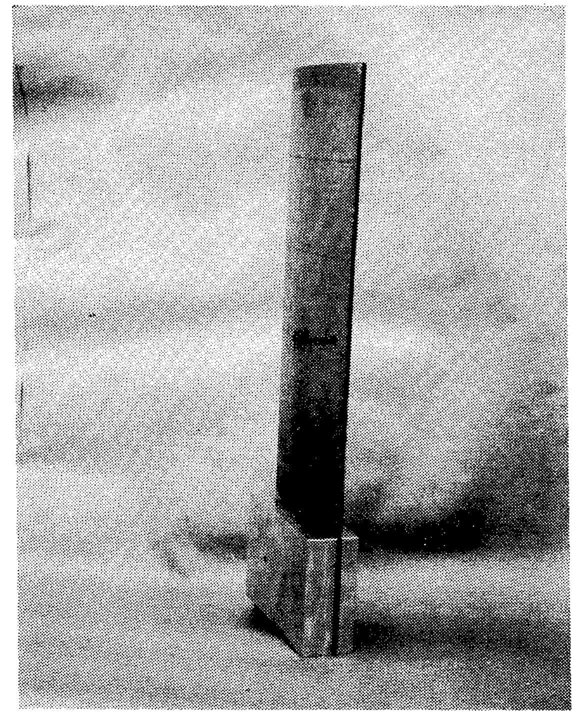


Figure 16 Curved Panel - Specimen 16B

Table IV FLEXIBILITY MATRIX ELEMENTS FOR

CURVED PANELS

SPEC. NO.	METHOD	INFLUENCE COEFFICIENTS AT THE TIP (IN./100 LB.)						AVERAGE % ERROR
		a(5,5)	a(5,10)	a(5,15)	a(10,10)	a(10,15)	a(15,15)	
15	DRR	0.90	.077	-.212	.200	.095	0.99	46.84
	EXP	1.37	.126	-.673	.242	.222	2.81	--
16*	DRR	18.60	16.14	13.84	14.81	13.32	12.77	--
16A	DRR	7.60	6.51	5.56	6.26	5.85	6.11	7.57
	USA	7.64	6.05	4.77	5.61	5.07	5.42	16.15
	EXP	8.74	7.54	6.08	6.46	6.26	6.08	--
16B	DRR	1.39	1.08	.846	1.18	1.16	1.49	28.60
	USA	1.58	0.863	.227	.934	.951	2.04	34.44
	EXP	2.03	1.30	.586	1.40	1.46	2.58	--

*Analysis of 16 as a flat plate for comparison purposes - not a test specimen.

The procedure was checked for composite plate stability with Procedure RA5 and compressive buckling of curved isotropic plates with Timoshenko [8]. Good agreement was obtained in both cases.

Compressive buckling of composite curved plates was correlated with an extensive test series especially instituted for this program. Shear buckling of curved plates was correlated with design development tests for the F-5 fuselage component.

3.2.1 Panel Compression Tests

The test panels and test fixture used in the compression tests were described previously in Section 3.1.1 and are shown in Figures 5-9.

Variations in the panels' curvature and warpage were slight and were corrected upon installation in the rigid loading fixture. Parallelism of loaded edges was determined on installation and corrected, where necessary, prior to a test run (parallelism to 0.003 in. over the edge length was assumed permissible).

Prior to assembly in the test fixture, each panel was bordered with Teflon tape, .003-inch thick, at all points that would be contacted by metal. This reduced the shear loads at the edges that resulted from high friction forces.

The structural similarity of the curved panel specimens was such that a reliable test procedure had to be developed and rigidly adhered to in order to clearly distinguish between the response of the various panels. To aid in this process, the same holding fixture, which accepted various panel thicknesses, was used in all vibration, pressure and buckling tests. A common procedure for installing the panels and aligning the set-up for test runs proved to be highly relevant in obtaining repeatable and satisfactory results. The salient features in installing the panel were to finger-tighten the bolts on the unloaded edge supports when simple support conditions were used, and wrench-tighten (to 60 in.-lb.) the bolts where clamped supports were used. In each case, the bolts were checked after two low-load excursions were applied (these loadings were used to seat the panel and remove most of the hysteresis).

Following the panel installation an axial load was applied using a 120,000-pound Baldwin Universal test machine.

In the buckling test, the information required was out-of-plane movement of the panel as the axial load was increased from 100 pounds to the critical load level. This movement was monitored by two methods: a linear differential transformer whose output was sent to a machine-mounted, x-y drum recorder and by the moire' shadow method.

The moire grid shadow method is an experimental procedure used to measure out-of-plane movements of a surface. Its principal advantage, especially for buckling tests, is that a full-field view of surface movements can be observed as the test progresses. A brief description of how the method works and the equipment used in its application on the panel studies are explained in the following paragraphs. The development of this procedure was based on the information obtained from Reference [9].

The essential pieces of equipment used in developing the moire patterns are a master grid pattern and a rigid transparent backing plate to hold the grid next to the panel. Locations of these elements on a typical test are sketched in Figure 17. In the experiments described in this report, a Kodak Carousel projector for the light source and a mounted plexiglas plate, formed to the same contour as the specimen, to hold the grid pattern in place was used. This is shown in Figures 18 and 19. With this set-up, the grid shadow was obliquely cast on the white surface of the panel. The observer, looking through the master grid, saw two grids superimposed, and as the panel points moved to, or away from, the master grid, the shadowed grid would move up or down by the amount

$$y = \delta \tan \alpha \tag{110}$$

When the panel deflected a distance equal to the pitch, ρ, of the master grid a dark band or fringe would appear. The shape and width of the fringe, as well as the number of fringes seen in an area were, therefore, a function of the change in curvature of the panel over the given area and the grid pitch. For example, a local buckle or a tight hump in the panel would display very narrow and closely spaced fringes, whereas an overall buckle would show very wide fringes which would be spaced far apart. On the other hand, if the grid pitch were halved, the sensitivity of the set-up would be doubled, or, twice as many fringes per unit deflection would be seen. The type of grid originally used in the buckling test was determined by assuming a

SMD7051

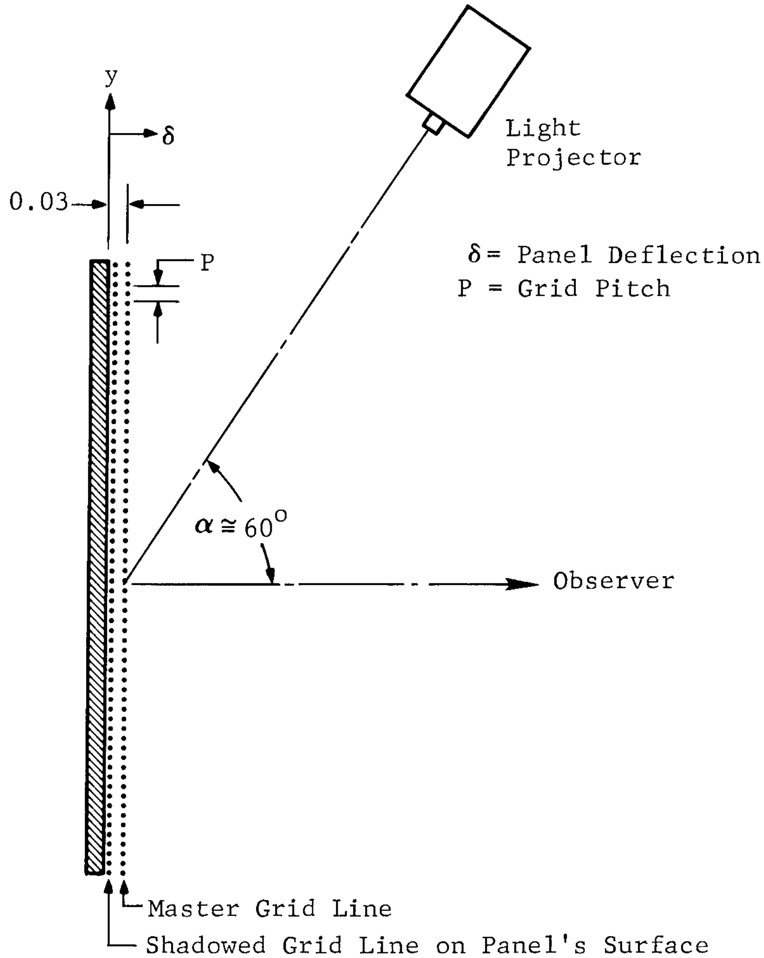


Figure 17 Test Set-Up Using the Moire Grid Shadow Method

SMD7052

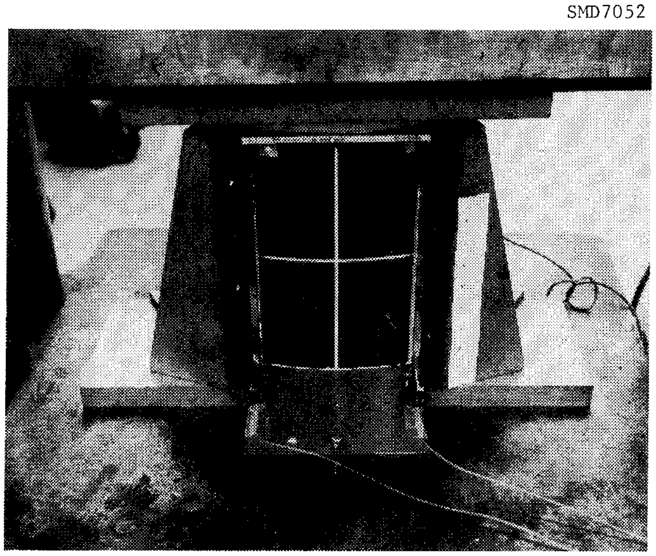


Figure 18 Rear View of Master Grid Plate and Support Structure

SMD7053

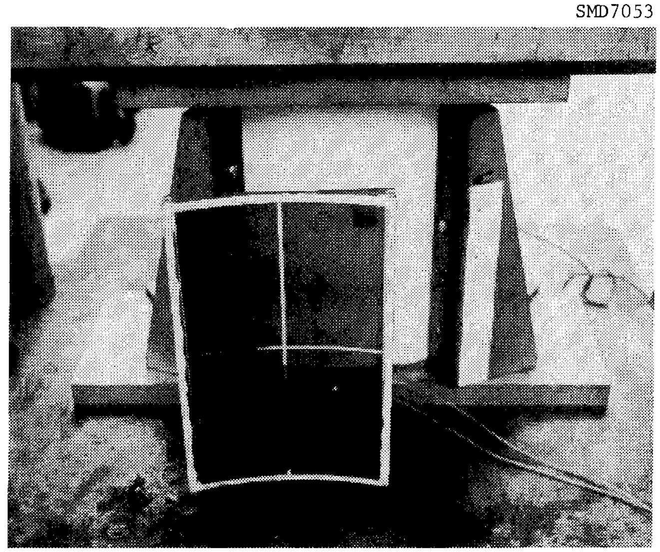


Figure 19 Master Grid as Mounted on Curved Plexiglass Surface

sensitivity of one fringe per 0.01-inch deflection would be desirable. Using the following equation

$$P = \delta \tan \alpha = 0.01 \tan 60^\circ \tag{111}$$

it was determined that 0.017 inch/grid lines, or approximately 50 lines per inch, would be acceptable. Buckling tests with this pattern showed promising results but a need for more sensitivity was required to obtain a better definition of the panel's deflection. Subsequent tests showed that grids having 100 lines/inch gave satisfactory results.

Upon installation, the differential transformer's plunger was lightly spring-loaded against the panel and displaced such that a null balance was achieved at the recorder. The location of the plunger relative to the panel was established by viewing the movements of the moire fringe pattern on the opposite face during the initial loadings. The area having the greatest fringe shift indicated the most out-of-plane activity, thus locating the plunger to obtain maximum deflections.

The moire patterns, which were developed on the white surface of the painted panel, were used to stop the loading when buckling was observed to be imminent. The characteristics of the pattern at this point were rapid fringe movement and the decreasing distance between adjacent fringes. When these conditions occurred, the load was immediately dumped and the maximum load attained was recorded.

The test setup and some representative moire photographs are shown in Figures 20 through 25. Many more photos are shown in Reference [7].

During the time the moire patterns were being observed, a simultaneous plot of the out-of-plane motion at an established point on the opposite panel face was made. This plot of deflection vs. load was provided by the test machine's integral recording system. These curves, an example of which is shown in Figure 26, were used to obtain Southwell plots (see Figure 27) which ultimately provided the critical buckling load of the panel. All Southwell plots are shown in Reference [7]. The Southwell method is a technique for obtaining the buckling load of a structure from experimental load-deflection information. The details of its implementation differ depending on the structure being

SMD7054

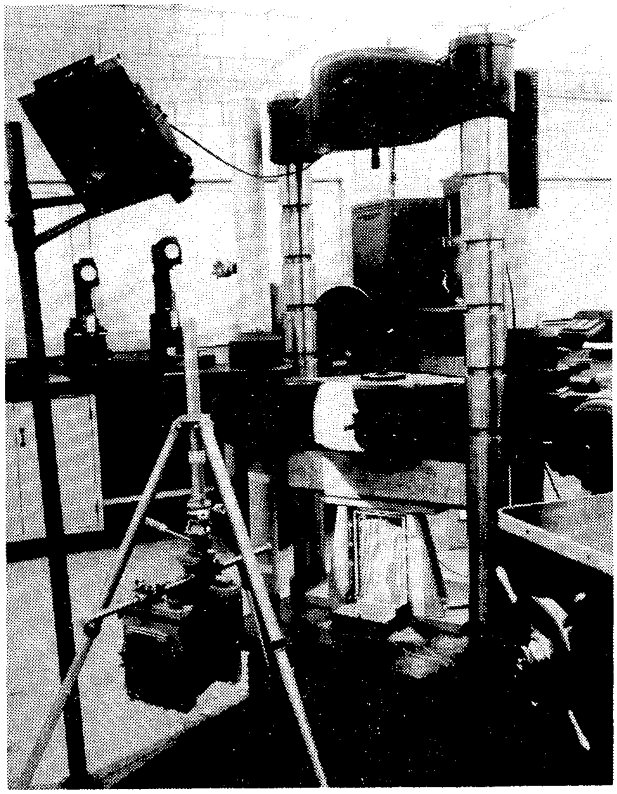


Figure 20 Test Setup for Buckling Investigation

SMD7055

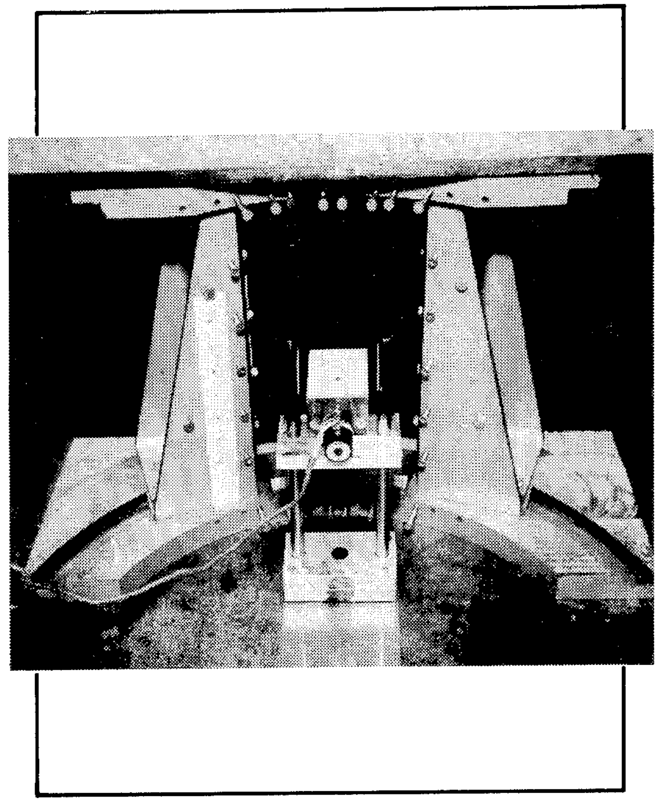


Figure 21 Rear View of Buckling Setup

SMD7056

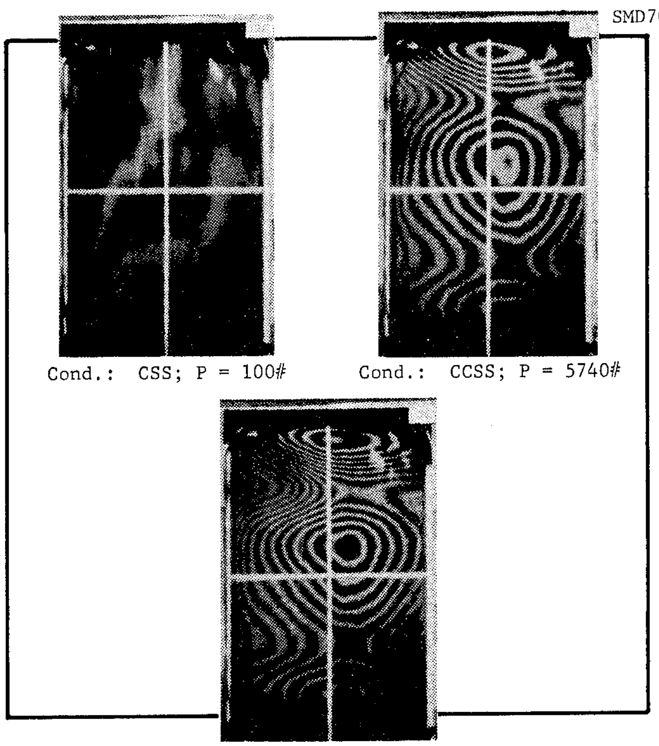


Figure 22 Moire Patterns for -19E

SMD7057

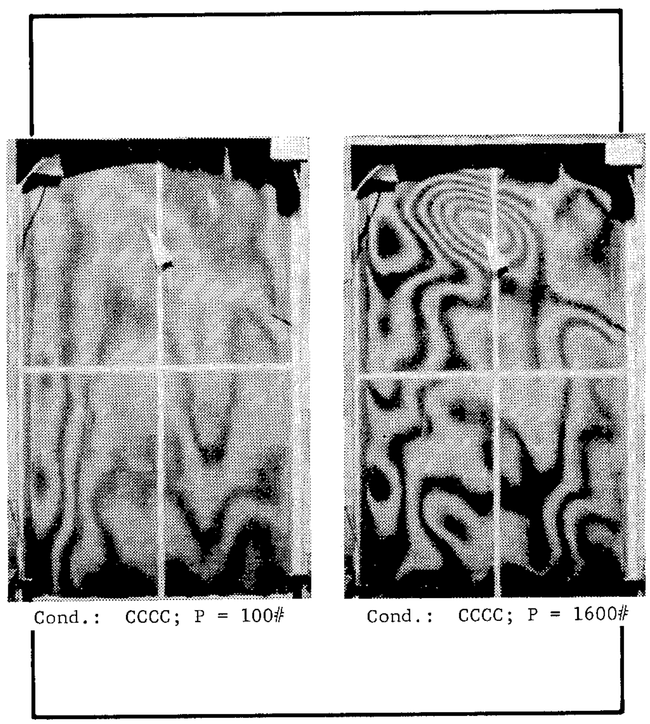


Figure 23 Moire Patterns for -23C

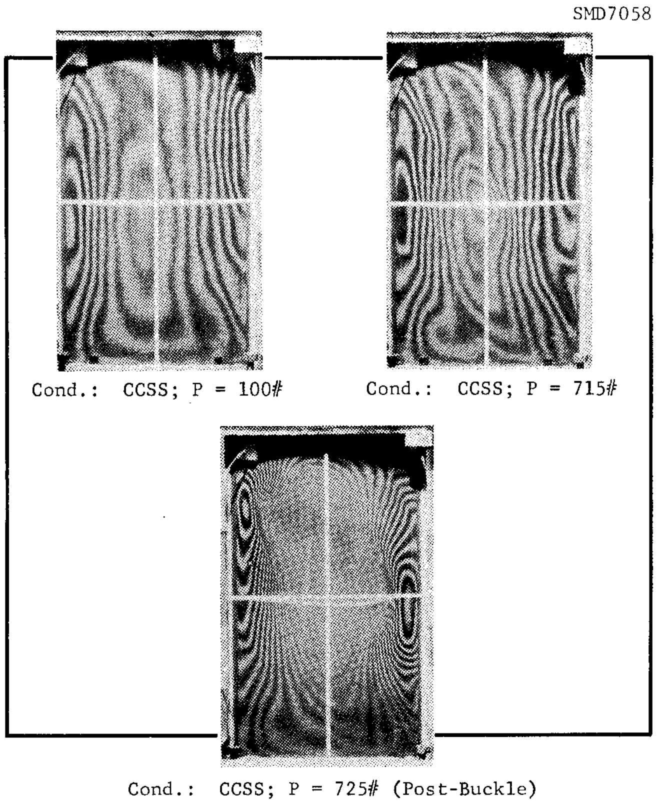


Figure 24 Moire Patterns for -37A

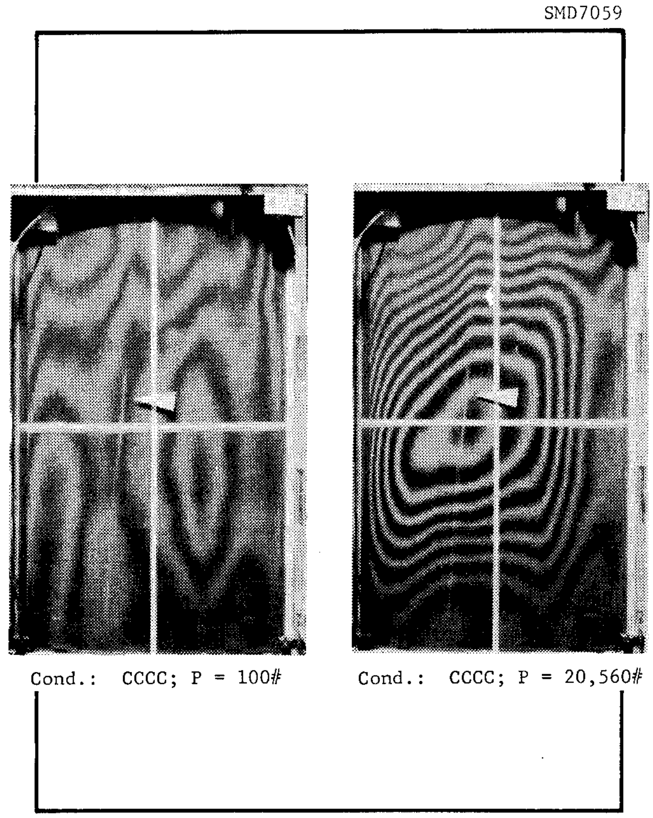
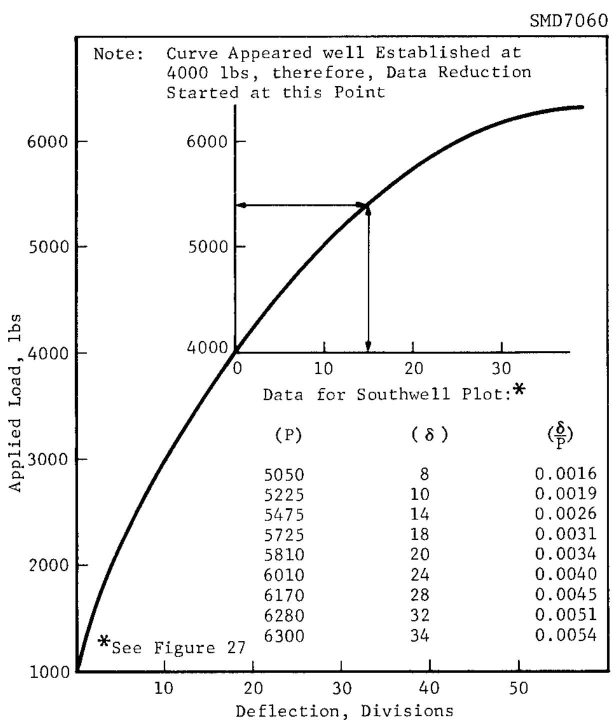


Figure 25 Moire Patterns for -47B



(P)	(δ)	(δ/P)
5050	8	0.0016
5225	10	0.0019
5475	14	0.0026
5725	18	0.0031
5810	20	0.0034
6010	24	0.0040
6170	28	0.0045
6280	32	0.0051
6300	34	0.0054

Figure 26 Typical Load-Deflection Curve

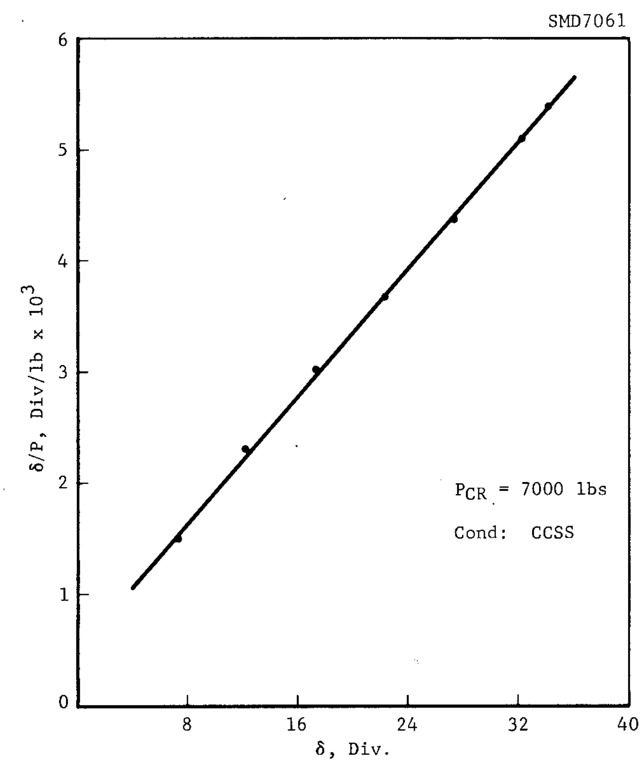


Figure 27 Southwell Curve for -33B

analyzed. It has been used for the buckling of columns, beam-columns, plates, and more recently, shells.

The theoretical basis for the use of the Southwell method for shells may be found in the works of Tenerelli and Horton [10] and Galletly and Reynolds [11] . A modification of the method of Tenerelli and Horton was used here.

Briefly, the moire grid-shadow method was used in the initial load cycle (below the buckling load) to find the point of maximum deflection on the shell. The linear variable differential transformer (LVDT) was then positioned to read deflections at that point. On subsequent load cycles, the load-deflection plot for that point was read out on the rotating drum of the test machine. A typical plot is shown in Figure 26.

The actual Southwell plots were generated by using a Hewlett-Packard 9100B calculator with a plotter. A program was written to take the load-deflection data as input and produce a plot of (deflection/applied load) versus deflection. Using the straight portion of this plot, the buckling load is calculated as the inverse of the slope of the line.

The moire procedure used in obtaining buckling loads of the various panels proved to be quite satisfactory and saved the majority of panels for future tests. There were, however, a number of panels that snapped into a post-yield buckle before loading could be stopped. When this condition occurred the panels were damaged to the extent that subsequent load cycles produced lower buckling loads. On the other hand, when the loads were dumped at initial evidence of buckling, subsequent loading cycles produced repeatable results. On a few panels, all three methods (moire, Southwell and snap-through) were used to obtain the critical buckling load. Comparing the results of these methods, using Table V, it can be seen that satisfactory correlation exists.

Curved aluminum panels were also tested to obtain base reference data for evaluating the edge restraints of the fixture. The results from these tests indicated that the clamping action on the loaded edges of the specimens was very near the classical value, however, the simple supports provided slightly more than classical restraint. This excess edge moment was 10 inch-pounds per radian per inch of length. This value was determined to be within acceptable limits and the tests proceeded without further alterations in set-up procedures.

TABLE V

BUCKLING RESULTS FOR GRAPHITE EPOXY COMPOSITE CURVED PANELS

PANEL NUMBER	LAMINATE IDENTI-FICATION	MEAN THICKNESS INCHES	THICKNESS MEAN DEVIATION INCHES	VERTICAL EDGES SIMPLY SUPPORTED, CURVED EDGES CLAMPED						ALL EDGES CLAMPED		
				SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.	SS8 LOAD LBS.	KNOCKDOWN FACTOR EXP	KNOCKDOWN FACTOR SS8	SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.
17A	$[0/90]_{2s}$	0.0592	0.0021		6680	7323	7200	.93	.80		6740	7088
17B	$[0/90]_{2s}$	0.0528	0.0036		4865		5900	.83	.73			
19A	$[\pm 45]_{2s}$	0.0696	0.0030		8660	8750	12400	.70				
19B	$[\pm 45]_{2s}$	0.0707	0.0030	9000		9050	12700	.71	.61			
19C	$[\pm 45]_{2s}$	0.0713	0.0025		8820		13000	.68	.62			
19D	$[\pm 45]_{2s}$	0.0719	0.0019		8760		13200	.66	.59			
19E	$[\pm 45]_{2s}$	0.0598	0.0026		5740		9500	.60	.44		5980	
21A	$[0/90]_{s}$	0.0289	0.0015		985	1125	1530	.64	.84	1195	1175	
21B	$[0/90]_{s}$	0.0282	0.0013		925		1470	.63	.83			
23A	$[\pm 45]_{s}$	0.0354	0.0025	1870	1870	1914	4000	.47				
23B	$[\pm 45]_{s}$	0.0362	0.0033	1610		1695	4180	.38	.35			
23C	$[\pm 45]_{s}$	0.0340	0.0018		1590	1624	3780	.42	.35		1625	
23D	$[\pm 45]_{s}$	0.0359	0.0025	1850			4130	.45	.47			
23E	$[\pm 45]_{s}$	0.0307	0.0013		1280	1314	2950	.43	.46			

TABLE V, Cont'd.

BUCKLING RESULTS FOR GRAPHITE-EPOXY COMPOSITE CURVED PANELS

PANEL NUMBER	LAMINATE IDENTI-FICATION	MEAN THICKNESS INCHES	THICKNESS MEAN DEVIATION INCHES	VERTICAL EDGES SIMPLY SUPPORTED, CURVED EDGES CLAMPED						ALL EDGES CLAMPED		
				SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.	SS8 LOAD LBS.	KNOCKDOWN FACTOR EXP	KNOCKDOWN FACTOR SS8	SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.
27	(Alum)	0.0630		17000			23500	.72	.48			
29C	$[\pm 45]_{3s}$	0.1045	0.0027		17760	23125	27700	.64	.63			
29D	$[\pm 45]_{3s}$	0.1066	0.0037			21889	29200	.62	.62			
29E	$[\pm 45]_{3s}$	0.0892	0.0040		10780		19700	.55	.52			
31A	$[+45]_{2s}$	0.0343	0.0024	1550	1550	1759	2800	.55	.16			
31B	$[+45]_{2s}$	0.0356	0.0038		1505	1704	2900	.52			1480	
31C	$[+45]_{2s}$	0.0353	0.0017		1520		2900	.52				
31D	$[+45]_{2s}$	0.0347	0.0021	1500			2800	.54				
31E	$[+45]_{2s}$	0.0289	0.0015	975			2000	.49	.33			
33A	$[+45]_{4s}$	0.0692	0.0031	7050			11700	.60				
33B	$[+45]_{4s}$	0.0679	0.0024	6340	6300	7000	11300	.56	.50			
33C	$[+45]_{4s}$	0.0622	0.0030		5700	5750	9300	.61	.24			
33D	$[+45]_{4s}$	0.0709	0.0035		6620		12200	.54	.29			
33E	$[+45]_{4s}$	0.0591	0.0024	4000	4000	4021	8300	.47				

TABLE V, Cont'd.

BUCKLING RESULTS FOR GRAPHITE-EPOXY COMPOSITE CURVED PANELS

PANEL NUMBER	LAMINATE IDENTI-FICATION	MEAN THICKNESS INCHES	THICKNESS MEAN DEVIATION INCHES	VERTICAL EDGES SIMPLY SUPPORTED, CURVED EDGES CLAMPED						ALL EDGES CLAMPED		
				SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.	SS8 LOAD LBS.	KNOCKDOWN FACTOR EXP	KNOCKDOWN FACTOR SS8	SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.
35A	$[+45]_{6s}$	0.0902	0.0049		9180	10270	20000	.46	.36			
37A	$[-30]_{2s}$	0.0282	0.0071	725	715		2200	.33	.41			
39A	$[-30]_{4s}$	0.0580	0.0022		4730		8000	.59	.73		4985	
41A	$[-30]_{6s}$	0.0900	0.0018		10460	10435	17800	.59	.84			
43A	$[0]_{2s}$	0.0364	0.0020		1315	1575	2100	.63	.68			
43C	$[0]_{2s}$	0.0368	0.0032	1540			2100	.73	.64			
43D	$[0]_{2s}$	0.0362	0.0024	1315	1290	1418	2100	.63	.64			
43E	$[0]_{2s}$	0.0294	0.0020	945			1800	.53	.49			
45A	$[0]_{4s}$	0.0701	0.0018		5580	6468	8700	.64	.67		7300	7704
45B	$[0]_{4s}$	0.0699	0.0028	5735			8700	.66	.62			
45C	$[0]_{4s}$	0.0696	0.0014		5300	5553	8700	.56	.66			
45D	$[0]_{4s}$	0.0695	0.0014		5080	5610	8700	.58	.66		6600	7123
45E	$[0]_{4s}$	0.0582	0.0029		5105	5122	5800	.88	.64			
47A	$[0]_{6s}$	0.1064	0.0030		16500	18362	21600	.76	.61			

TABLE V, Cont'd.

BUCKLING RESULTS FOR GRAPHITE-EPOXY COMPOSITE CURVED PANELS

PANEL NUMBER	LAMINATE IDENTI-FICATION	MEAN THICKNESS INCHES	THICKNESS MEAN DEVIATION INCHES	VERTICAL EDGES SIMPLY SUPPORTED, CURVED EDGES CLAMPED						ALL EDGES CLAMPED		
				SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.	SS8 LOAD LBS.	KNOCKDOWN FACTOR EXP	KNOCKDOWN FACTOR SS8	SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.
47B	$[0]_{6s}$	0.1039	0.0027		18000	19598	20600	.85	.62		20560	21538
47C	$[0]_{6s}$	0.1013	0.0035		16760	17812	19600	.56	.59			
49A	$[0/90]_{3s}$	0.0880	0.0026		14680	16625	16200	.91	.79			
49B	$[0/90]_{3s}$	0.0781	0.0034		12460	14118	12500	.99	.79			
51A	$[\mp 30]_{s}$	0.0296	0.0019		1150		2630	.44	.55			
53A	$[\mp 30]_{2s}$	0.0557	0.0023		5405	5818	7750	.70	.70			
55A	$[\mp 30]_{3s}$	0.0807	0.0026		12900	13860	17000	.76	.68			

TABLE V, Cont'd.

BUCKLING RESULTS FOR GRAPHITE-EPOXY COMPOSITE CURVED PANELS

PANEL NUMBER	LAMINATE IDENTIFICATION	MEAN THICKNESS INCHES	THICKNESS MEAN DEVIATION INCHES	VERTICAL EDGES SIMPLY SUPPORTED CURVED EDGES CLAMPED: SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.	SS8 LOAD LBS.	KNOCKDOWN FACTOR: EXP	SS8	ALL EDGES CLAMPED: SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.
57A	$[0/-45/90/+45]_s$	0.0574	0.0018		8240	8966	10000	.82	.90			
57B	$[0/-45/90/+45]_s$	0.0499	0.0028		6640	6691	7500	.86	.80			
57C	$[0/-45/90/+45]_s$	0.0516	0.0023		6460	6897	7900	.82	.85		7100	
57D	$[0/-45/90/+45]_s$	0.0499	0.0028		5820	6416	7500	.78	.80			
57E	$[0/-45/90/+45]_s$	0.0524	0.0032		6960	7194	8250	.84	.78			
59A	$[0/\mp 60]_s$	0.0422	0.0010		3355	3595	5200	.64	.92		3730	3846
59B	$[0/\mp 60]_s$	0.0392	0.0018		3390	3626	4450	.76	.84	3685	3530	
59C	$[0/\mp 60]_s$	0.0382	0.0026		3400	3582	4200	.81	.79			
59D	$[0/\mp 60]_s$	0.0397	0.0020		3000	3170	4600	.65	.82			
59E	$[0/\mp 60]_s$	0.0390	0.0028		3460	3846	4420	.78	.76			
61A	$[0/\mp 60]_{2s}$	0.0870	0.0026		22950	23871	27400	.84	.76			
61B	$[0/\mp 60]_{2s}$	0.0794	0.0041		18080	18571	22800	.79	.70			
61C	$[0/\mp 60]_{2s}$	0.0785	0.0034		16920	18136	22300	.76	.71			

TABLE V, Concluded

BUCKLING RESULTS FOR GRAPHITE-EPOXY COMPOSITE CURVED PANELS

PANEL NUMBER	LAMINATE IDENTIFICATION	MEAN THICKNESS INCHES	THICKNESS MEAN DEVIATION INCHES	VERTICAL EDGES SIMPLY SUPPORTED CURVED EDGES CLAMPED						ALL EDGES CLAMPED		
				SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.	SS8 LOAD LBS.	KNOCKDOWN FACTOR EXP	KNOCKDOWN FACTOR SS8	SNAP LOAD LBS.	MOIRE' LOAD LBS.	SOUTHWELL LOAD LBS.
61D	$[0/\mp 60]_{2s}$	0.0782	0.0029		18800	19000	22300	.84	.74			
67	Alum	0.0320		3825	3645		8500	.45				
69A	$[0_2/\mp 45]_s$	0.0512	0.0026		5500	5663	8150	.68	.63			
69B	$[0_2/\mp 45]_s$	0.0521	0.0019	5410	5410	5114	8150	.62	.70			
69C	$[0_2/\mp 45]_s$	0.0488	0.0028		5385	5604	7400	.73	.61			
69D	$[0_2/\mp 45]_s$	0.0504	0.0025		5310	5581	7900	.67	.63			
69E	$[0_2/\mp 45]_s$	0.0506	0.0034	5870	5700	5882	8000	.73	.58			
71A	$[0/\mp 45]_s$	0.0408	0.0021		2930	3187	4870	.60	.75			
71B	$[0/\mp 45]_s$	0.0394	0.0019		2595	2803	4540	.57	.76			
71C	$[0/\mp 45]_s$	0.0394	0.0021	2810	2810	2910	4540	.62	.75			
71D	$[0/\mp 45]_s$	0.0397	0.0020		2610	2942	4600	.57	.75			
71E	$[0/\mp 45]_s$	0.0390	0.0025		2310	3333	4440	.70	.71			

During the course of the buckling tests, a documentary film was generated showing the installation and testing of a typical graphite panel. This film, which is retained in the Composite Structures Engineering Group, provides a graphic display of the moire pattern development as the panel was loaded and the onset of buckling.

Table V includes the SS8 classical buckling load predictions for each panel as well as the knockdown factor predicted by SS8 based on the standard deviation in thickness of each panel. For the aluminum panels, the knockdown factor is found from the equation (Reference [50])

$$\gamma = 1 - 0.901\left(1 - e^{-\frac{1}{16}\sqrt{R/t}}\right) \tag{112}$$

Although the knockdown factor based on the standard deviations of the panel thickness is not always conservative, it does indicate trends fairly well and should be investigated further.

Figure 28 is a summary of all the buckling data obtained in terms of the ratio between experimental and classical buckling load versus R/t. In Figures 29 through 38, the results according to laminate orientation are separated to show that some types of laminates seem to be much more sensitive to imperfections than others and that the thin laminates are the most sensitive.

3.2.2 Panel Shear Tests

Two test specimens, one graphite-epoxy and one boron-epoxy, were used in this investigation. Each specimen consisted of an assembly of four quarter-circle panels nine inches long on a 12-inch radius, as shown in Figure 39. In both cases the basic test panels consisted of eight plies oriented at ± 45 degrees to the cylinder axis.

Loads were applied to the test apparatus to produce pure torsion. Strain gages were installed on both the inner and outer surfaces of each panel. Electrical deflection gages were installed inside the specimens to record radial deflection of the panels.

Testing was directed toward (1) the determination of the buckling stress and (2) the examination of the post-buckling strength. Determination of buckling stresses required loading the specimens to 75-90 percent of the buckling load. This

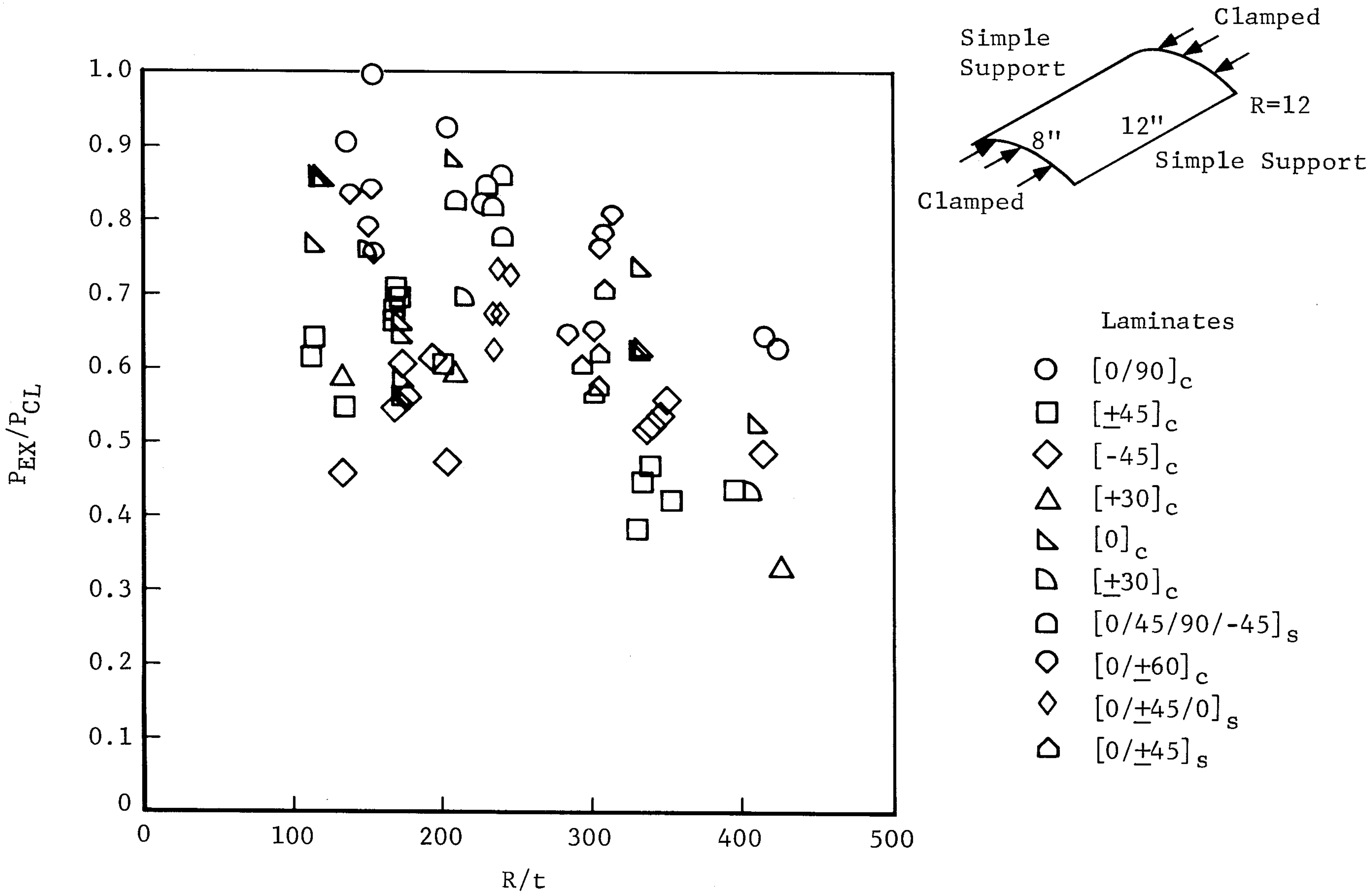


Figure 28 Curved Panel Buckling Summary

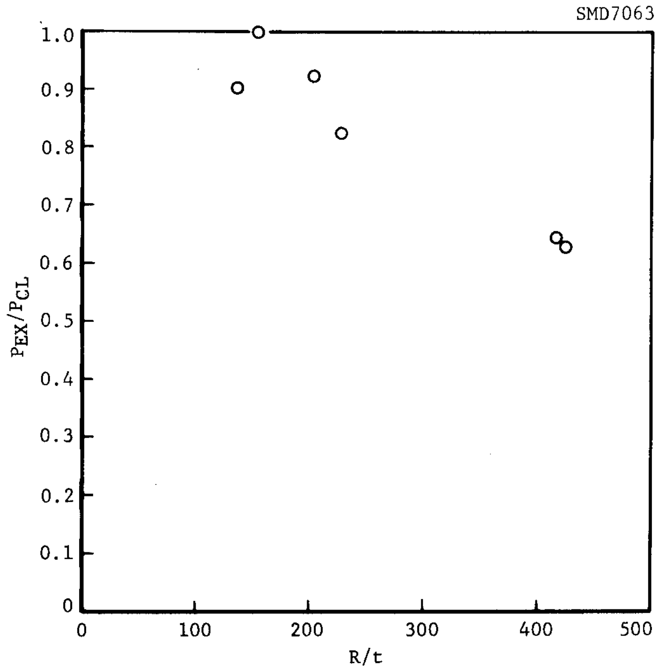


Figure 29 Curved Panel Buckling Plot: $[0/90]_c$

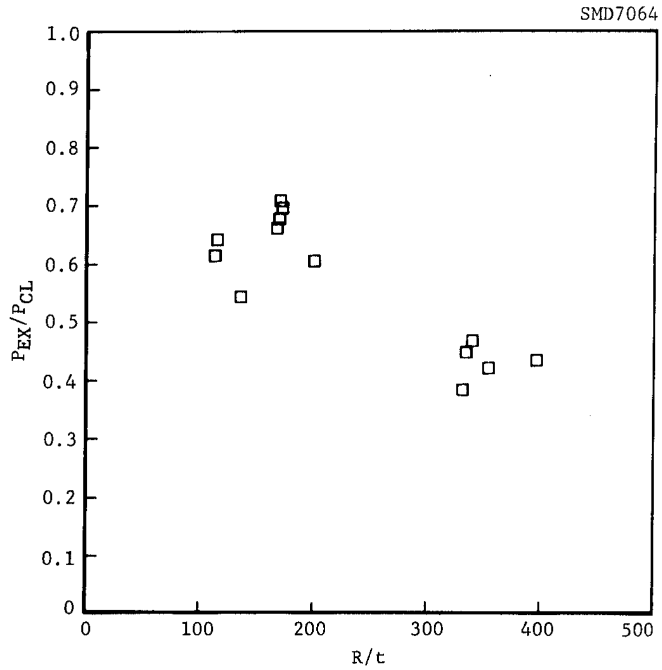


Figure 30 Curved Panel Buckling Plot: $[\pm 45]_c$

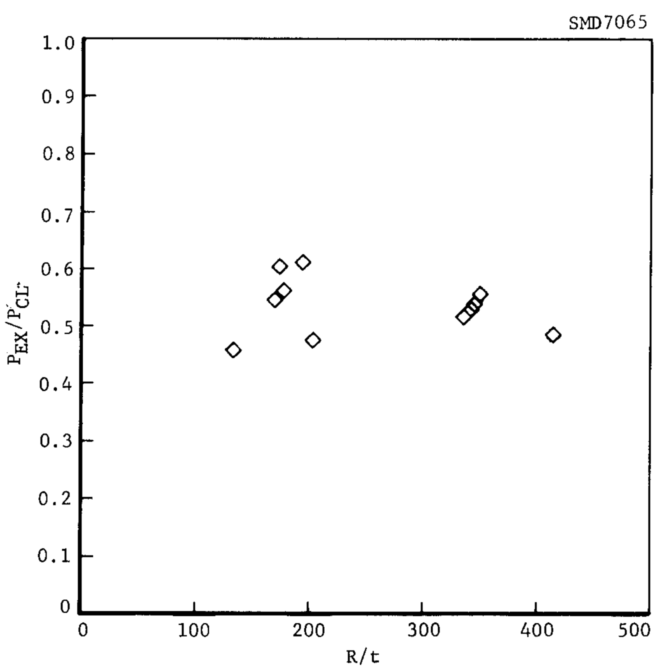


Figure 31 Curved Panel Buckling Plot: $[-45]_c$

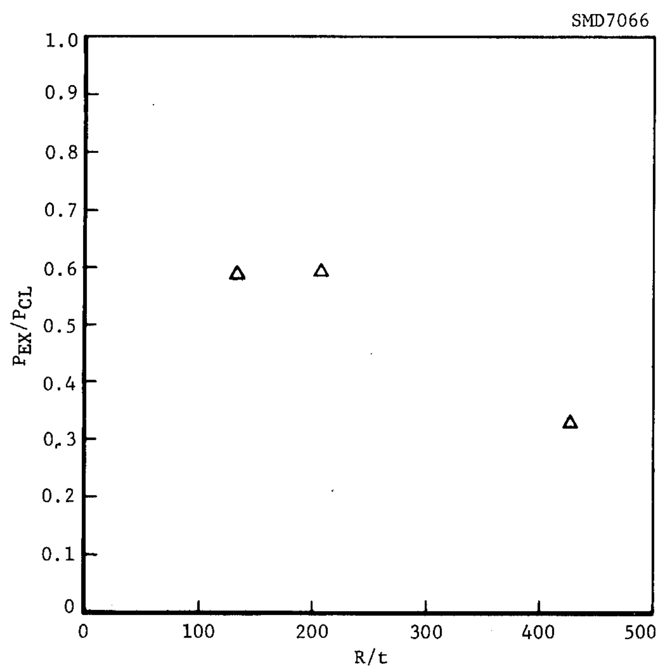


Figure 32 Curved Panel Buckling Plot: $[+30]_c$

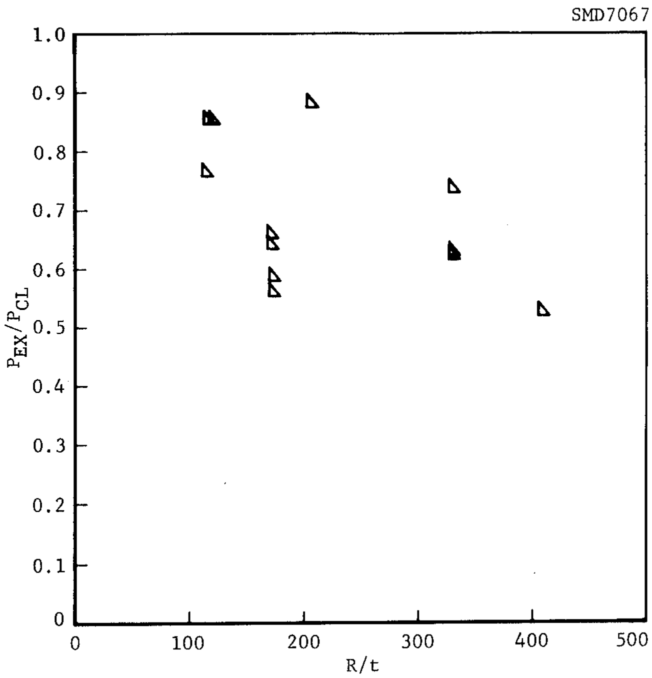


Figure 33 Curved Panel Buckling Plot: $[0]_c$

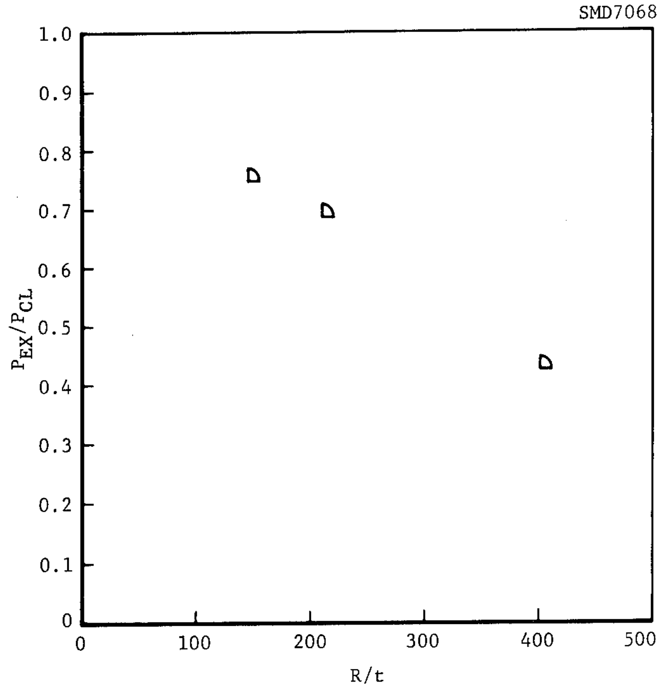


Figure 34 Curved Panel Buckling Plot: $[\pm 30]_c$

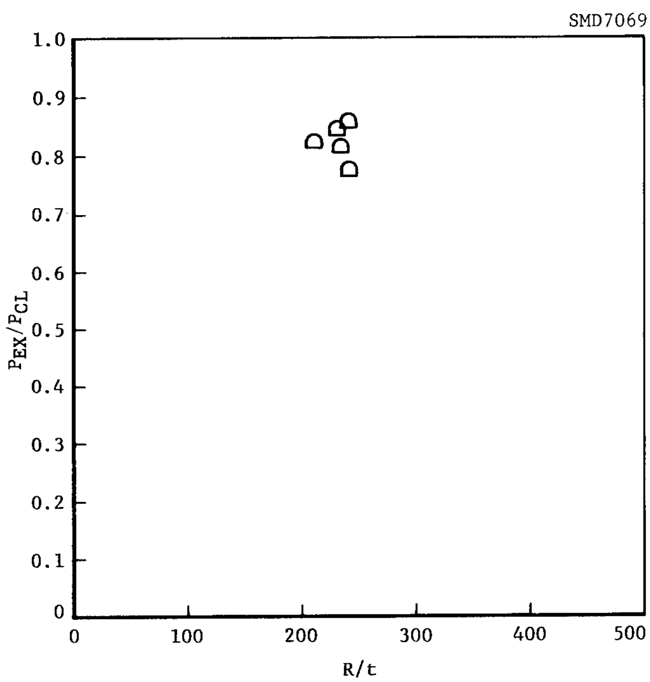


Figure 35 Curved Panel Buckling Plot: $[0/45/90/-45]_s$

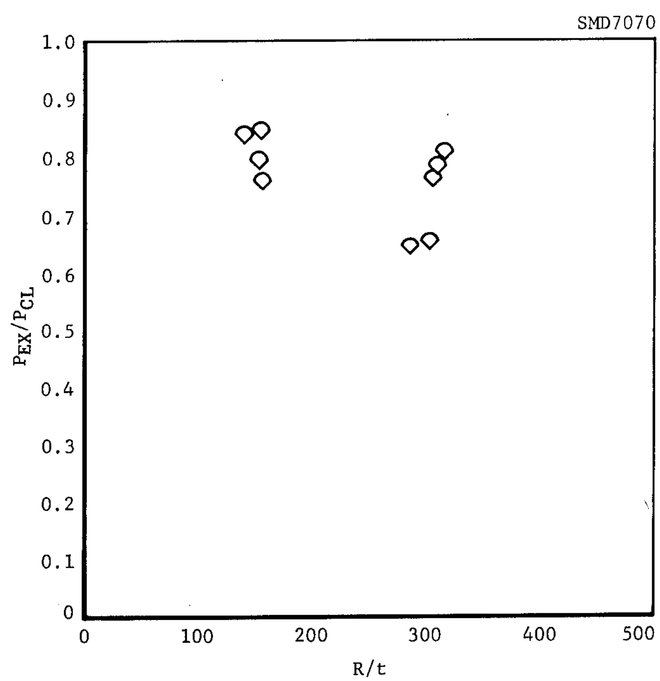


Figure 36 Curved Panel Buckling Plot: $[0/\pm 60]_c$

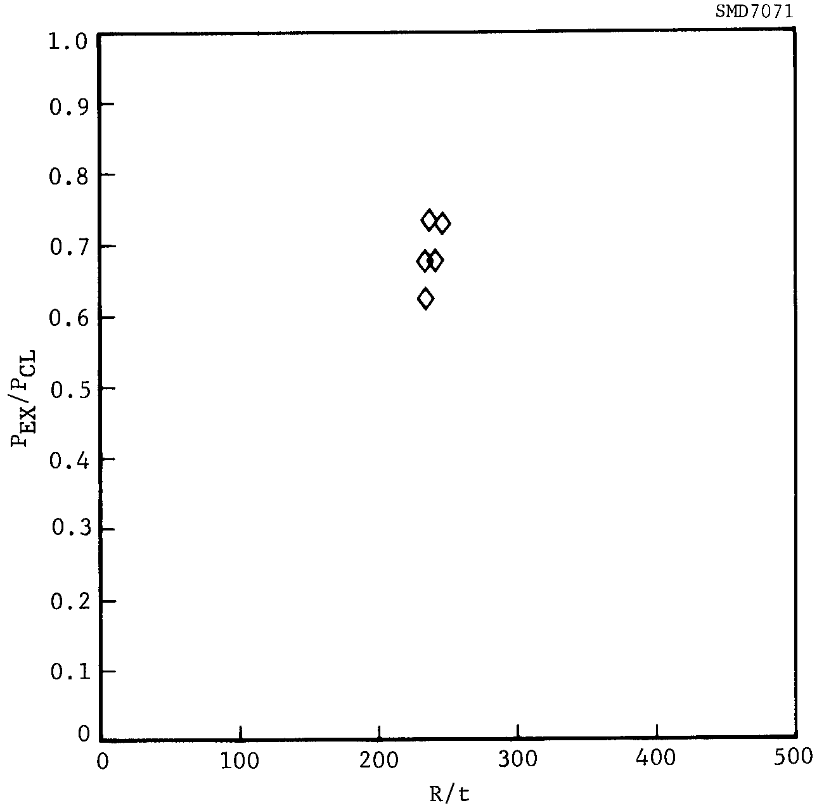


Figure 37 Curved Panel Buckling Plot: $[0/\pm45/0]_s$

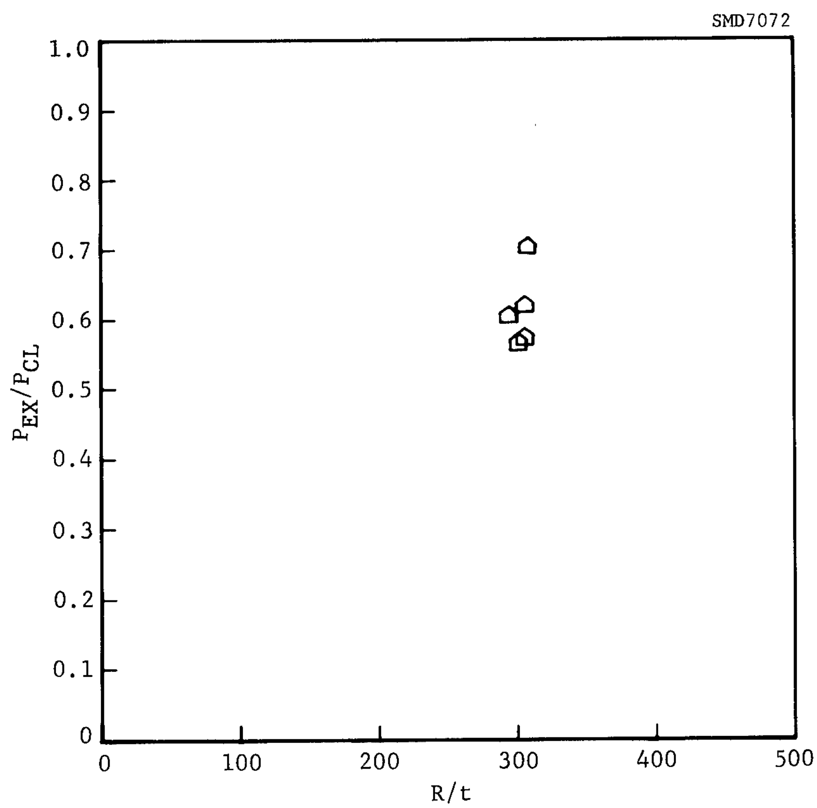


Figure 38 Curved Panel Buckling Plot: $[0/\pm45]_s$

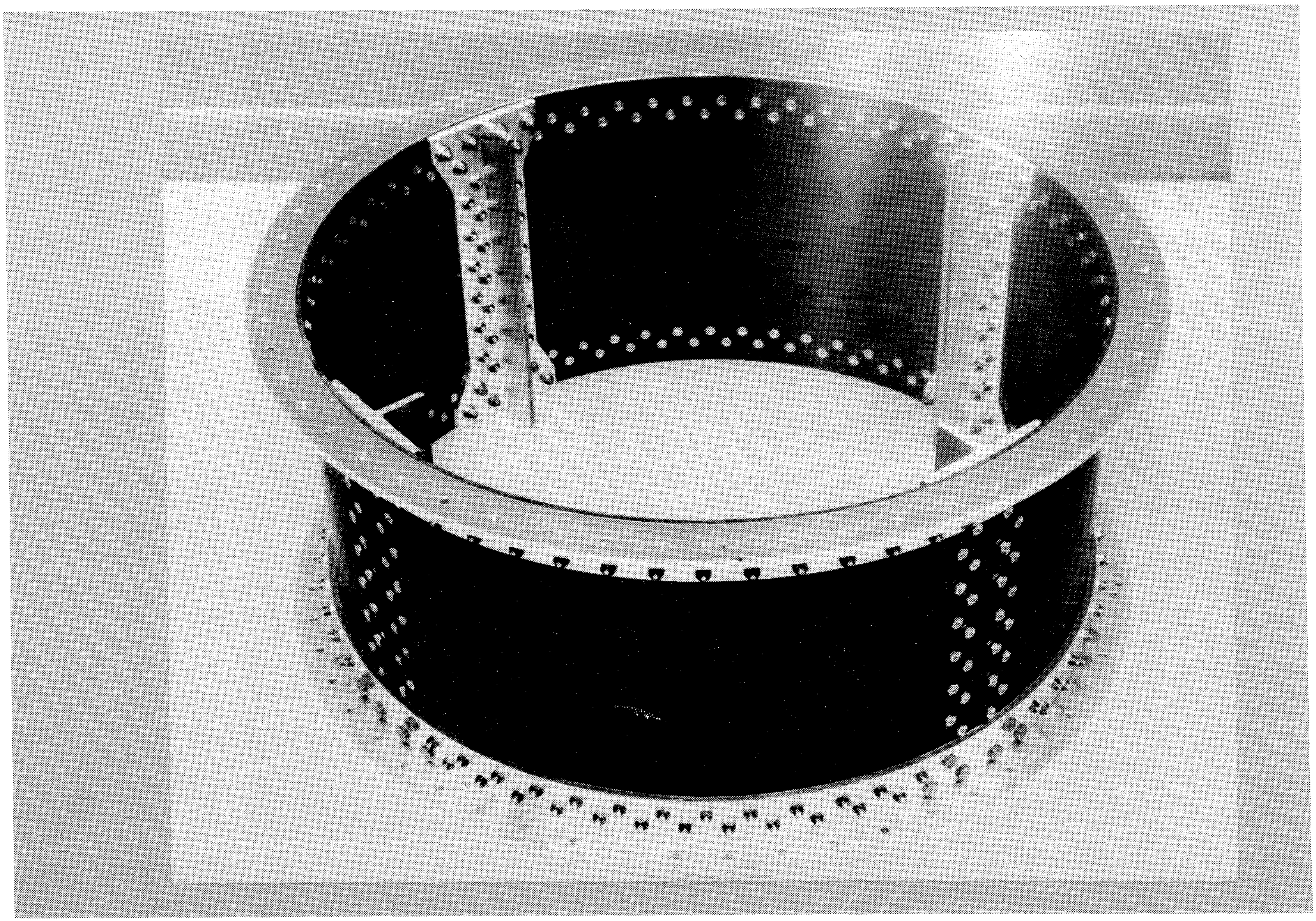


Figure 39 Curved Panel Test Assembly

procedure was usually repeated several times to check the repeatability of the data. This process was repeated for both directions of applied torque to determine the buckling stresses for different directions of applied shear. Because the composite panels consist of relatively few plies, stacking results in a basic imbalance with respect to laminate bending. This results in significantly different values of shear buckling stress of the panel for opposite directions of shear application.

Buckling stresses were experimentally determined through a "modifed" Southwell Method which is a logical extension of the works of Galletly and Reynolds [11] , and of Horton and Craig [12] . This method requires loading only near the actual buckling load which is desirable since actually buckling the test specimen could cause local damage and affect subsequent results. Moreover, the method allows use of the more reliable strain gages as opposed to deflection gages.

This method utilizes the stress (or load) versus surface strain curve from any point on the buckle at loads approaching buckling. This curve becomes increasingly nonlinear as the buckling load is approached, because of the increase of local bending at the buckle. The departure from linearity in terms of strain is defined as $\Delta\epsilon$. According to Galletly and Reynolds [11] , the buckling stress (load) is equal to the inverse slope of the $\Delta\epsilon/P$ versus $\Delta\epsilon$ curve, (P may denote either load or stress). This technique was applied to all strain data taken during these tests with generally good results. Values of ϵ much below 100 μin./in. generally give unreliable results because of the sensitivity limits on instrumentation. Typical results are shown below in the separate discussions of each test.

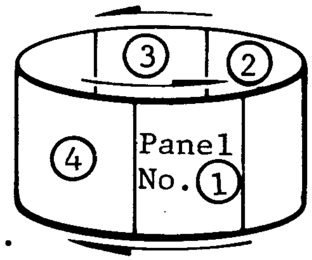
Results of the buckling tests on the graphite-epoxy panels are summarized in Table VI. Data was obtained from each of the four panels composing the test assembly. Conditions 1 and 2 refer to different directions of shear. As is seen, a significant difference in buckling loads results.

Strain gage rosettes placed back-to-back in pairs were used to obtain the data for the buckling stress determination. These gages were also used to compute K_{xy} which relates the shear stress on each panel τ to the total load P applied to the test assembly. These gages, with the exception of 1 and 2 (Table VI) are located at the center of the panel. Gages 1 and 2 were located in a corner near the edge.

Table VI GRAPHITE-EPOXY-CURVED PANEL SHEAR BUCKLING RESULTS

8 Ply $\pm 45^o$ Laminate

Average Thickness = .056 in.



		Condition 1 $\tau_{xy} > 0$			Condition 2 $\tau_{xy} < 0$				
			Southwell			Southwell		Deflection	
Panel No.	Position	K_{xy} (psi/1b)	P_{CR} (1b)	τ_{CR} (psi)	K_{xy} psi/1b	P_{CR} (1b)	τ_{CR} (psi)	P_{CR} (1b)	τ_{CR} (psi)
1	1	.445	-	-	.475	27,325	12,960	-	-
1	2	.445	-	-	.475	-	-	-	-
1	3	.445	18,431	8210	.475	26,667	12,670	26,000	12,350
1	4	.445	18,434	8210	.475	26,730	12,710	-	-
2	5	.476	-	-	.475	26,774	12,720	27,000	12,800
2	6	.476	-	-	.475	26,570	12,610	-	-
3	7	.476	19,020	9060	.481	-	-	26,500	12,750
3	8	.476	18,400	8770	.481	-	-	-	-
4	9	.516	18,950	9780	.508	-	-	24,000	12,200
4	10	.516	18,730	9690	.508	-	-	-	-
		Theory τ_{CR} = 9170 psi CL-CL; = 7420 psi SS-SS			Theory τ_{CR} = 13,670 psi CL-CL; = 10,720 psi SS-SS				

Deflection gages were placed to monitor lateral deflection at the panel center. Because of their low sensitivity these gages did not record any appreciable deflection until the panels actually buckled and very large deflections resulted. This behavior is shown in Figure 40. The points at which the deflections became large are those values listed in Table VI. These values support the Southwell data very well.

Theoretical buckling stresses were determined for the case of clamped edges and simply supported edges. The values (Table VI) for the clamped edges agree very well with the experimental results. Actual edge conditions approach the clamped case because of the stiff edgemembers and ample mechanical fasteners used.

An example of typical data used in the Southwell determination is shown in Figure 41. This data was taken for Condition 1 at Panel 4. The associated Southwell plots are seen in Figure 42.

Post-buckling behavior of the graphite-epoxy panels was characterized by large deflections with several buckles visible in each panel (see Figure 43). The behavior in terms of deflection is illustrated in Figure 2. Buckling of each panel occurred in sequence with a load drop accompanying each. With all four panels buckled, only a small amount of additional load was carried (20%) before failure. Failure occurred catastrophically at an average panel stress of approximately 16,000 psi as typically shown in Figures 44 and 45.

Results of the boron-epoxy buckling tests are summarized in Table VII. As before, loads and buckling stresses for both directions of loading were obtained with back-to-back strain gages on each of the four panels.

The data is seen to be very consistent. Analysis is seen to agree favorably as before although the results approach the case of simply-supported edges.

Typical load-strain curves and the associated Southwell plot for the boron panels are shown in Figures 46 and 47.

Visual and photographic observations of the post buckling behavior revealed that cracks appeared very soon after buckling occurred. Very little additional load was carried beyond buckling. The highest load attained was 21,500 pounds while buckling occurred at 19,000 to 20,000 pounds.

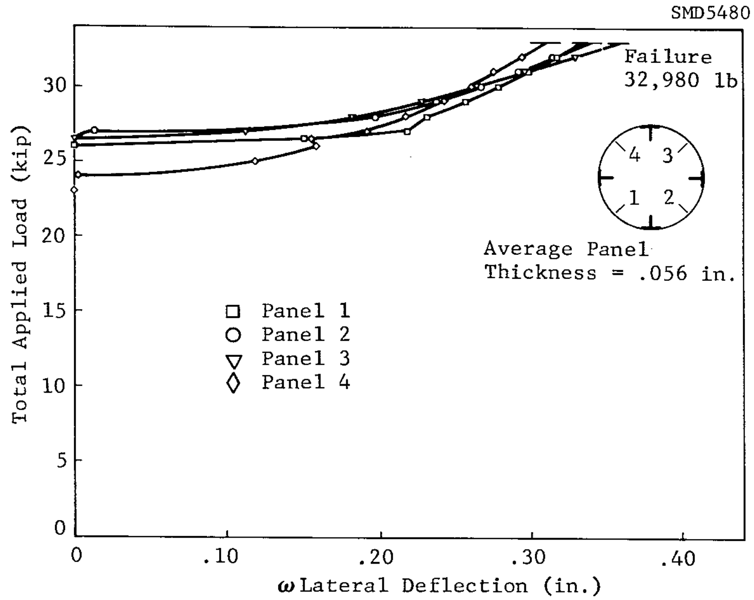
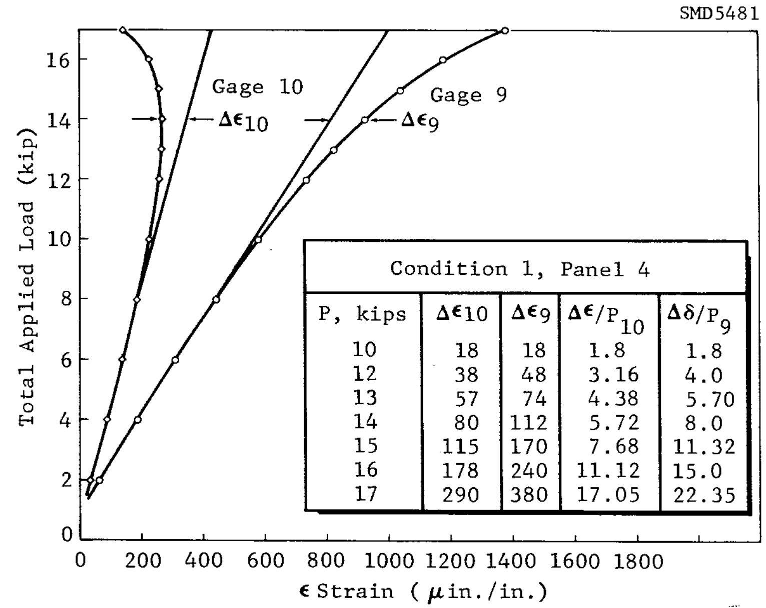


Figure 40 Load Versus Deflection for Graphite Shell



Condition 1, Panel 4

P, kips	$\Delta\epsilon_{10}$	$\Delta\epsilon_9$	$\Delta\epsilon/P_{10}$	$\Delta\delta/P_9$
10	18	18	1.8	1.8
12	38	48	3.16	4.0
13	57	74	4.38	5.70
14	80	112	5.72	8.0
15	115	170	7.68	11.32
16	178	240	11.12	15.0
17	290	380	17.05	22.35

Figure 41 Load Versus Strain-Graphite Curved Panel Test

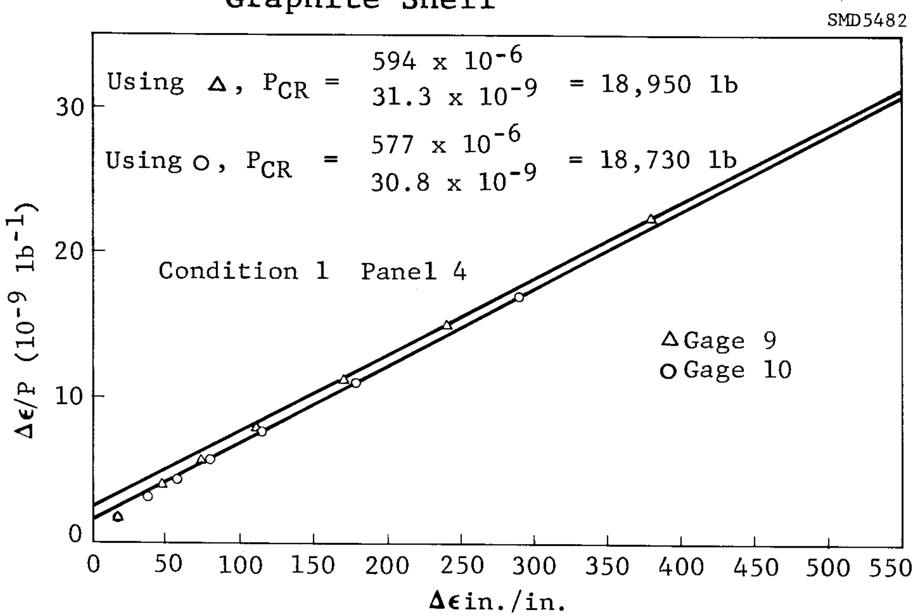


Figure 42 Southwell Plot - Graphite Curved Panel

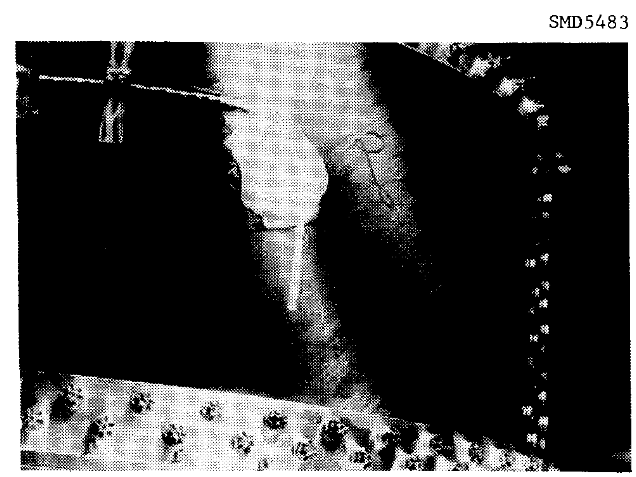


Figure 43 Graphite-Epoxy Curved Panel After Buckling

SMD5484

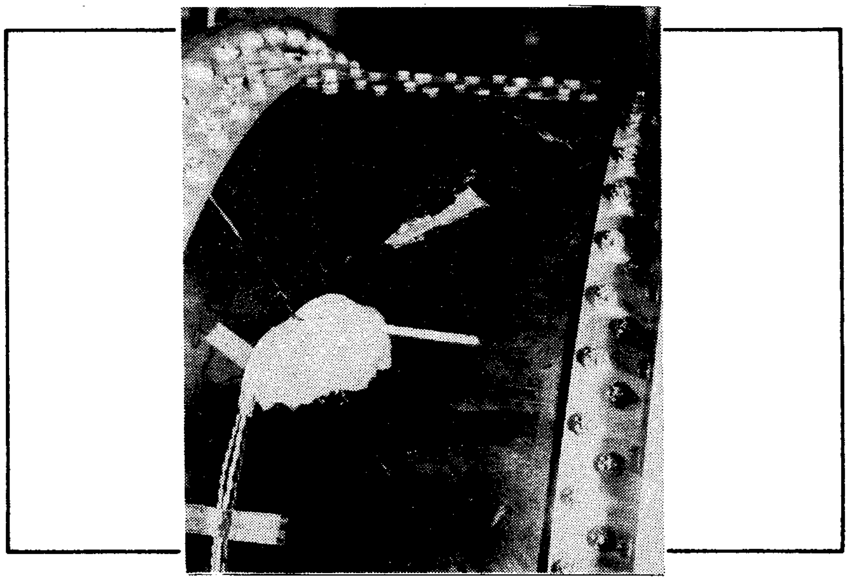


Figure 44 Failed Graphite-Epoxy Curved Panel

SMD5485

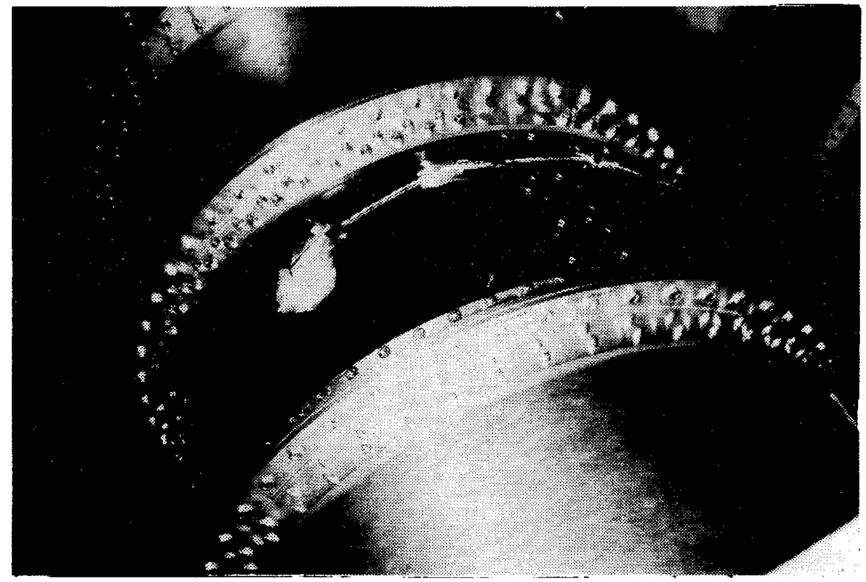


Figure 45 Failed Graphite-Epoxy Curved Panel - Overall View

SMD5486

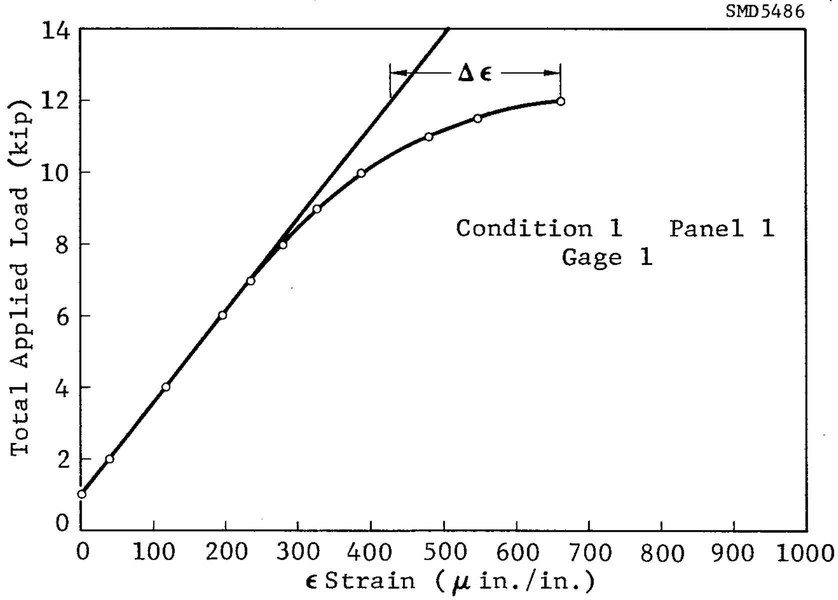


Figure 46 Load Versus Strain-Boron Curved Panel

SMD5487

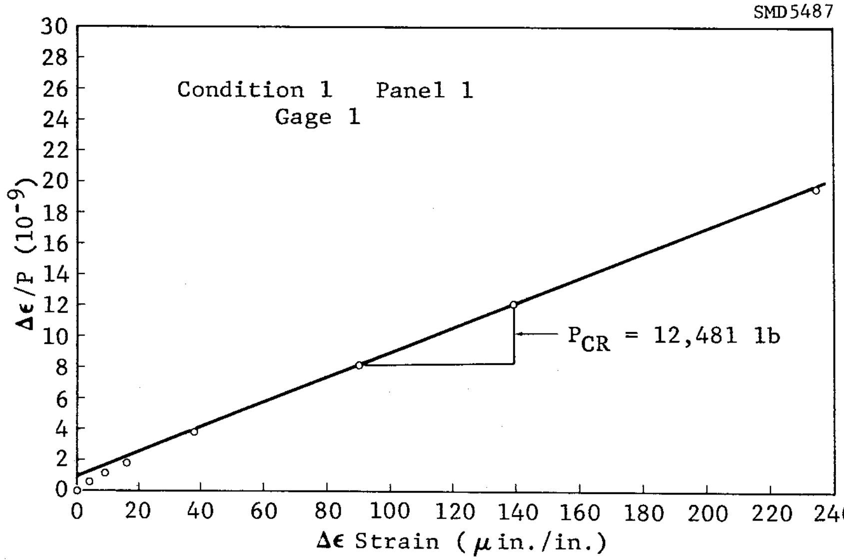
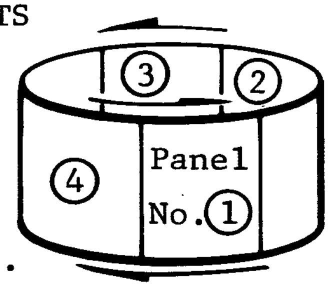


Figure 47 Southwell Plot - Boron Curved Panel

Table VII BORON-EPOXY CURVED PANEL SHEAR BUCKLING RESULTS



8 Ply $\pm 45^{o}$ Laminate   Average Thickness = .042 in.

		Condition 1 $\tau_{xy} > 0$			Condition 2 $\tau_{xy} < 0$				
			Southwell			Southwell		Deflection	
Panel No.	Position	K_{xy} (psi/lb)	P_{CR} (lb)	τ_{CR} (psi)	K_{xy} psi/lb	P_{CR} (lb)	τ_{CR} (psi)	P_{CR} (lb)	τ_{CR} (psi)
1	1	.613	12,481	7650	.608	20,154	12,220	20,250	12,300
1	2	.613	12,424	7620	.608	19,468	11,820	-	-
2	3	.603	13,742	8290	.595	19,862	11,810	20,250	12,050
2	4	.603	14,078	8470	.595	19,665	11,700	-	-
3	5	.598	13,864	8290	.592	19,595	11,600	19,800	11,720
3	6	.598	14,190	8480	.592	19,091	11,300	-	-
4	7	.594	-	-	.592	19,855	11,750	20,700	12,280
4	8	.594	-	-	.592	19,760	11,710	-	-
		Theory τ_{CR} = 9140 psi CL-CL = 7600 psi SS-SS			Theory τ_{CR} = 13,290 psi CL-CL = 10,480 psi SS-SS				

3.3 VIBRATION

The vibration option was run extensively in checkout of SS8. Again, the anisotropic plate capability was checked with RA5 and showed good agreement.

The work of Sewall (Reference [14]) was used to compare natural frequency data for isotropic curved panels. As an example of the type of correlation obtained, the following results were obtained for an aluminum panel with a = 11.0 inches, b = 9.0 inches, t = 0.028 inch, and R = 48.0 inches. For one longitudinal and two circumferential modes, the following results were obtained:

	f, cps
SS8, simply supported edges	180.0
Sewall analysis, simply supported edges	184.0
SS8, clamped edges	468.6
Sewall analysis, clamped edges	536.5

The results indicate that SS8 gives a better frequency estimate than Sewall's analysis, since an energy solution gives an upper bound for the frequency, and SS8 shows a lower frequency in both cases. This is to be expected because Sewall neglected modal coupling effects in his one-term Rayleigh-type analysis.

For isotropic cylinders, the results of Park, et al. (Reference [15]), were used for comparison. They tested a steel cylinder built in at one end and free at the other. The dimensions were a = 48.0 inches, R = 10.0 inches, and t = 0.03 inches. They found the lowest natural frequency at m = 1, n = 4 of 50.4 cps. SS8 predicts a value of 51.9 cps. For m = 1, n = 3, the experimental value was 51.5cps., while SS8 predicts 55.3 cps. For m = 1, n = 5, the experimental value was 70.9 cps., while SS8 predicts 71.5 cps.

The anisotropic capability of SS8 was tested by comparing its results with those of Bert, et al. (Reference [16]), who presented exact analytical solutions for the natural frequencies of anisotropic simply-supported cylinders. As an example, they studied a two-layer, cross-ply cylinder using material properties typical of boron-epoxy. Some examples of the excellent agreement obtained are shown below.

SS8	Ref. (11)	M	N
235 cps	235 cps	1	2
254 cps	253 cps	1	3
443 cps	443 cps	2	3

The dynamics of a cylinder with four internal stringers has been investigated and these investigations are documented in References [17], [18], and [19]. The SS8 results for this case show its discrete stiffener capability.

	SS8 Anal.	Ref. 17 Expt.	Ref. 18 Anal.	Ref. 19 Anal.
M = 1, N = 3	163 cps		158 cps	159 cps
M = 1, N = 4	99 cps	100 cps	99 cps	100 cps
M = 1, N = 5	91 cps	87 cps	91 cps	93 cps
M = 1, N = 6	106 cps	104 cps	105 cps	115 cps

Many other sources, References [20] - [47], were consulted for analytical and experimental information. Detailed correlation with these sources was not attempted since the layered composite capability could best be explored further through our test program.

3.3.1 Fuselage Program Tests

The specimens and fixture used for the Fuselage Program tests were described in Sections 3.1.1 and 3.2.1. The setup of equipment for the vibration tests is shown in Figures 48 and 49. The panel specimens were tested in the fully clamped boundary condition.

In the vibration tests the axial load was maintained at 100 pounds while the panel was tapped with a cardboard cylinder to set the panel vibrating at its resonant frequency. This frequency was monitored by the following equipment. The transducer was a one gram MB Electric Velocity Pickup (Model 115) connected to a Tektronix, Storage type Oscilloscope (Model 549). Incorporated in the system was a Krohn-Hite Variable Band Filter to obtain the frequency output within the ranges of interest. Photographs of the oscilloscope traces were made with a Hewlett-Packard Camera (Model 197A). These photographs constituted the data output of the system.

SMD7073

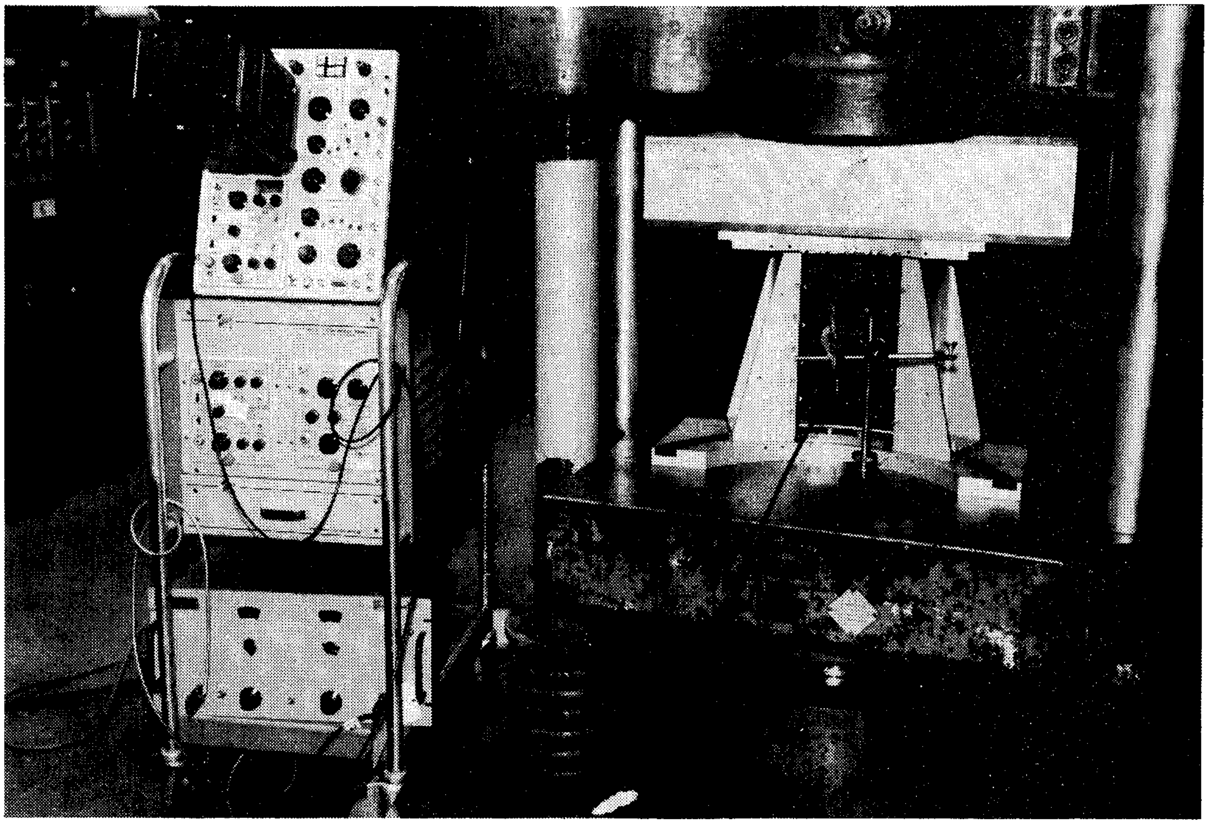


Figure 48 Test Setup and Instrumentation for Vibration Tests

SMD7074

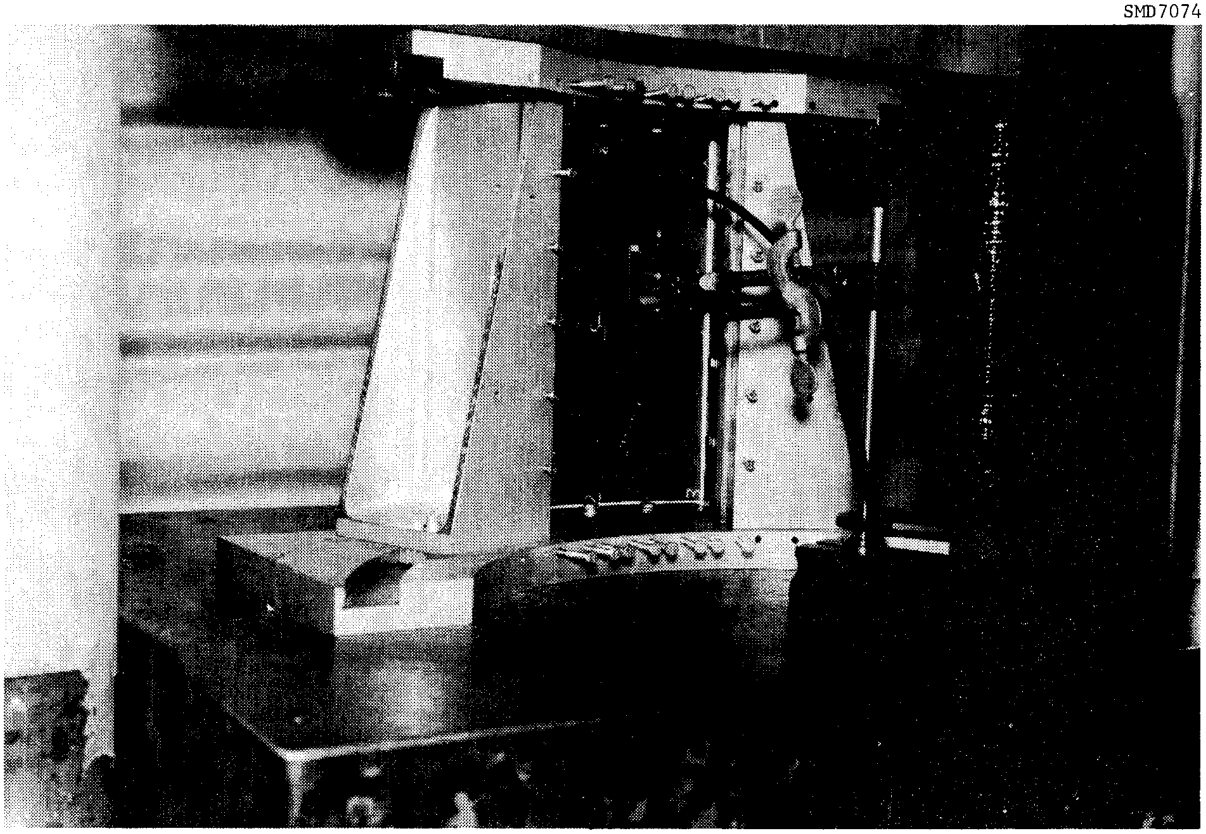


Figure 49 Vibration Setup Showing Closeup of Velocity Transducer

Typical photographs obtained during the vibration tests are included in Figure 50. All the photographs are given in Reference [7]. The fundamental frequency was obtained from these pictures using the following conversion formula:

$$\omega_o = \frac{N}{dRK}, \text{ cycles/record}$$

where: ω_o = fundamental frequency,

N = number of cycles counted,

d = distance on photograph to include N cycles,

R = ratio of object to image size to correct for photographic reproduction, and

K = constant set in on oscilloscope, seconds/cm

The actual process for measuring the distances on the photograph and converting the results, was accomplished on the Hewlett-Packard Data Reduction equipment. The final results are tabulated in Table VIII. The table shows panel number, laminate, the percent difference between experimental and results obtained using a 10 in.-lb./rad/in. elastic restraint on the straight sides, and the natural frequencies, including the clamped curved edge, simply supported straight edge classical results. The results show that the actual side support restraint makes a great deal of difference in the results.

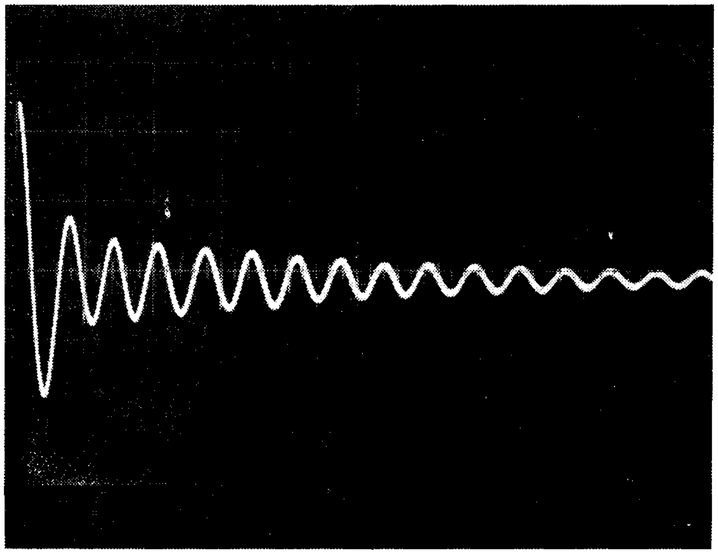
3.3.2 Dynamic Characteristics Program Tests

Some of the tests of Reference [13] were described in Section 3.1.2. The program also included tests of stiffened curved panels, unstiffened cylinders, and a stiffened cylinder.

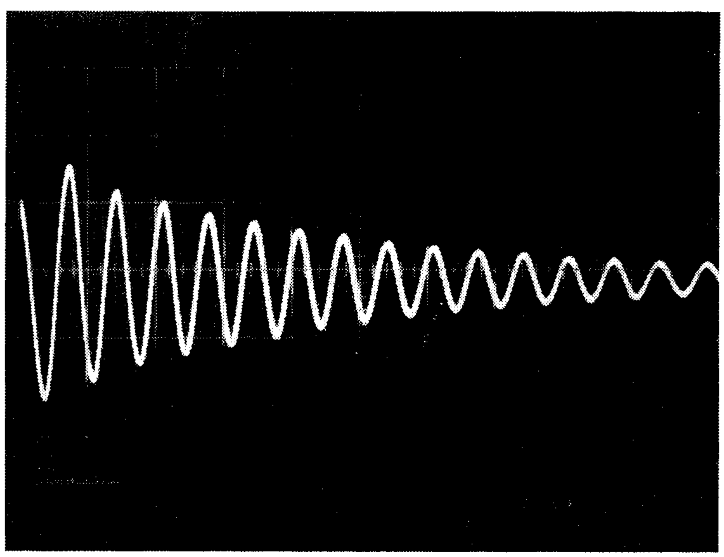
3.3.2.1 Cantilever Curved Panels

The specimens are described in Section 3.1.2 and shown in Figures 14-16. The specimens are designated 15, 16A, and 16B and have 15-inch spans and 24-ply, $[0/+45_4/90]_c$ laminates. Specimen 15 has a 15-inch chord and a 36-inch radius, while Specimens 16A and 16B have 6-inch chords and 36- and 12-inch radii, respectively.

Frequencies and mode shapes have been determined experimentally for the first six natural modes.



Panel 49A, Mode 1, 1



Panel 49A, Mode 1, 2

Figure 50 Velocity Traces for Panel 49A

Table VIII FUSELAGE PROGRAM VIBRATION TEST RESULTS

PANEL	LAYUP	% DIFF EXP-E.R.	MODE 1 NATURAL FREQUENCIES, HZ EXPER.	MODE 1 ELAST. RES. C-C-ER-ER	MODE 1 CLASSICAL CCSS	MODE 2 EXPER.	MODE 2 ELAST. RES. C-C-ER-ER	MODE 2 CLASSICAL CCSS
19A	$[\pm 45]_{2s}$	+ 1.3	771 (1, 2)	781	821			
19D	$[\pm 45]_{2s}$	+ 2.2	772 (1, 2)	789	831			
21A	$[0,90]_{s}$	+13.1	335 (1, 3)	379	411	415 (1, 2)	342	392
23E	$[\pm 45]_{s}$	+ 7.4	486 (1, 3)	522	544			
29E	$[\pm 45]_{3s}$	+12.3	729 (1, 2)	819	927			
33E	$[\pm 45]_{4s}$	+ 4.5	583 (1, 2)	609	665			
35A	$[\pm 45]_{6s}$	+ 7.4	702 (1, 2)	754	849			
39A	$[-30]_{4s}$	+ 8.1	595 (1, 2)	643	697	635 (1, 3)	726	780
41A	$[-30]_{6s}$	0	707 (1, 2)	707	836			
45E	$[0]_{4s}$	- 5.4	423 (1, 2)	400	467			
49A	$[0,90]_{3s}$	+ 2.6	707 (1, 2)	720	780	708 (1, 1)	633	782
51A	$[\mp 30]_{s}$	+ 8.2	451 (1, 3)	488	497			
53A	$[\mp 30]_{2s}$	+ 6.5	634 (1, 3)	675	739	637 (1, 2)	683	739
55A	$[\mp 30]_{3s}$	+11.9	649 (1, 2)	726	830			
59A	$[0,\mp 60]_{s}$	+ 8.9	514 (1, 3)	560	605	573 (1, 2)	569	613

Preliminary analyses were performed with the DRR curved panel analysis procedure (SS8). Post-test analyses were performed with the USA procedure and NASTRAN. All of the analyses included stacking sequence effects. The test-theory correlation data for natural frequencies is shown in Table IX. As seen in the table, the DRR analysis is in good agreement for the bending modes, which are dominated by the spanwise stiffness. However, the effect of curvature on the torsional stiffness is evidently being over-predicted in each case, thereby raising the frequencies for the torsion modes. Although several possible causes for the discrepancies have been investigated, no satisfactory explanation has yet been found for the failure of the DRR procedure to correctly model the torsional stiffness.

The opposite is true for the finite-element procedures. That is, the USA and NASTRAN analyses of Specimen 16B are modeling the torsional stiffness accurately, but they are overestimating the spanwise stiffness. Both simulations used piecewise flat element systems to model the structure. The torsional modes are not greatly affected by the curvature, but the curvature effects dominate the bending deflections. Therefore, the discrepancies reflect an inadequate representation of the specimen curvature. The superiority of the USA analysis to the NASTRAN analysis is caused by the larger number of elements used.

The agreement for Specimen 16A was greatly improved for both the DRR and USA analyses. The DRR analysis overpredicted the first torsional frequency, and the USA analysis overpredicted the bending stiffness for the fundamental mode and the influence coefficients. The superior agreement is caused by the relatively narrow chord and low curvature.

The USA analysis of Specimen 15 follows the previously noted trends in that it correctly predicts the first torsional frequency and accurately predicts all of the mode shapes. In this case, the first bending mode frequency and all subsequent frequencies were predicted to be lower than measured. The simulation used was an equivalent thickness and stiffness sandwich model with 11 spars and 16 ribs. Skin elements were flat, constant stress triangles. Agreement is not as good as it is for Specimen 16A although the curvatures are the same. The increased chord width and included angle increased curvature effects and made the specimen more difficult to analyze with flat elements.

Also included in the results is a DRR analysis of Specimen 16 as a flat panel for comparison purposes. Percent differences for 16A and 16B are shown to demonstrate the effect of curvature.

Table IX NATURAL FREQUENCIES FOR CURVED PANELS

SPEC. NO.	METHOD	FREQUENCY (Hz) 1	2	3	4	5	6	AVERAGE % ERROR
15	Mode	T	B	T	C	C	C	
	DRR	86.0	107	226	293	313	459	21.6
	EXP	61.0	94	178	246	271	405	--
16*	Mode	B	T	B	T	B	C	
	DRR	18.2	101	127	292	385	550	--
16A	Mode	B	T	B	T	B	C	
	DRR	27.8	123	163	344	438	609	4.57
	USA	29.2	110	155	319	396	551	5.97
	EXP	26.7	107	163	330	435	589	
16B	Mode	B	T	B	T	C	B	
	DRR	64.5	187	345	428	754	790	17.81
	USA	71.1	116	320	362	633	667	7.78
	NAST	81.5	114	418	373	708	846	12.93
	EXP	60.0	113	337	364	675	773	--

Modes: T = Torsion, B = Bending, C = Coupled

*Analysis of 16 as a flat plate for comparison purposes - not a test specimen.

3.3.2.2 Stiffened Panels

Free-free natural frequencies and mode shapes were measured for four stiffened panels, Specimens 33 through 36. One flat panel and one curved panel were fabricated. Each panel was 18 inches wide and 36 inches long, and the curved panel had a radius of 36 inches. Each panel was made of 12 plies of boron-epoxy oriented at $\pm 45/90$ degrees, resulting in plate bending stiffnesses $D_{11} = 155$, $D_{22} = 330$, and $D_{66} = 116$ lb.-in^2/in. Specimens 33 and 35 (curved) have three aluminum channel stiffeners bonded to one side at the interior quarter points.

Each stiffener has a cross-sectional area of 0.07625 in.^{2} and EI = 9508 lb-in.^{2} about the centroid. Specimens 34 and 36 were made by bonding two additional stiffeners to the edges of Specimens 33 and 35 after they were tested. The specimens were suspended horizontally with surgical tubing attached to one side along the panel length; this tubing was located nine inches from each end. The rigid body frequencies of the panel were one Hz or less.

To determine the validity of the experimental boundary conditions, Specimen 33 was also tested with the panel suspended vertically. The supports were attached to one end and were located five inches from each side. Frequencies and mode shapes were the same as those measured with the panel suspended horizontally.

Available analytical and test results for the stiffened panels are given in Table X. DRR results are shown for the flat panels, Specimens 33 and 34, and for the same panel without stiffeners for comparison. Acceptable analytical results for the curved specimens were not generated because of problems with the DRR shell analysis procedure SS8. Experimental results are shown for the lowest seven to nine natural frequencies detected. Analytical results for the flat plate are not complete in that some higher mode shapes had frequencies lower than some of those shown. Agreement was excellent between the experimental and analytical natural frequencies and mode shapes for the flat stiffened panels.

3.3.2.3 Unstiffened Composite Cylinders

Two unstiffened cylinders, 15 inches in diameter and 16 inches in length, were designed to study the accuracy of the Rayleigh-Ritz shell procedure SS8 for full cylinders.

Table X NATURAL FREQUENCIES (Hz) FOR STIFFENED PANELS

MODE	FLAT PLATE	SPECIMEN 33			SPECIMEN 34			SPEC 35	SPEC 36
	DRR	EXP	DRR	P.E.	EXP	DRR	P.E.	EXP	EXP
2,0	8.8	39.8	40.1	0.8	48.4	48.0	-0.8	--	--
3,0	25.1	95.7	100	4.5	126	129	2.4	--	--
4,0	48.9	--	162	--	--	250	--	--	--
1,1	17.0	18.1	17.1	-5.5	16.2	15.1	-6.8	19.6	18.3
2,1	36.0	48.2	48.3	0.2	59.2	59.6	0.7	78.0	76.0
3,1	61.7	--	101	--	--	146	--	--	169.4
4,1	87.7	--	155	--	--	271	--	--	--
0,2	57.8	54.5	54.3	-0.4	41.5	43.7	5.3	53.2	43.5
1,2	67.8	61.4	63.6	3.6	54.0	53.4	-1.1	63.8	56.4
2,2	92.9	93.6	93.2	-0.4	105.3	97.6	-8.2	84.8	82.6
3,2	129.2	--	157	--	--	189	--	140.8	--
0,3	158.9	--	143	--	120	122	1.7	144.1	121.6
1,3	164.2	--	152	--	142	130	-8.4	--	--
2,3	193.2	--	188	--	--	164	--	--	198.3
AVERAGE P.E.		--	--	2.2	--	--	3.9	--	--

Specimen 37 has six plies of boron-epoxy oriented at 0/±45 degrees, and Specimen 38 has four plies of boron-epoxy oriented at ±45 degrees. Frequencies, mode shapes, and damping coefficients were determined for the natural modes of the specimen corresponding to longitudinal mode m = 0, 1, 2 and the frequency sweep from 0 to 525 Hz. The specimens were tested with free-free boundary conditions as shown in Figures 51-53.

The frequency correlations are shown in Table XI and Figures 54 and 55. The actual cylinder properties and the predicted properties are given in Table XII. Agreement is good everywhere except the m = 2 modes for Specimen 38. Although the lamina modulus in the fiber direction was increased from 30 x 10^6 psi to 33.7 x 10^6 psi to account for the apparent high fiber volume fraction, the reduction in thickness of the shell brought about a 10 percent lower longitudinal stiffness than predicted. This resulted in lower frequencies.

3.3.2.4 Stiffened Composite Cylinders

Specimen 39, which is the graphite-epoxy stiffened shell shown in Figure 56, and in Figure 57 with an unstiffened cylinder, was fabricated and dynamic tested to study the accuracy of the DRR procedure SS8 for shells with stiffeners. This specimen is 24 inches in diameter and 30 inches in length with an 8-ply graphite shell with orientations of ±45 degrees. The plate bending stiffnesses for the shell are $D_{11} = D_{22} = 97$ and $D_{66} = 80$ lb-in^2/in. The shell is stiffened by four equally spaced aluminum external longitudinal stringers with EI = 2.3 x 10^6 lb-in.^{2}, two graphite internal rings at one-third and two-thirds of the length with EI = 6.4 x 10^5 lb-in^2, and two aluminum external rings at the ends with EI = 1.6 x 10^6 lb-in^2. Stiffener EI's were calculated about their centroids.

Attempts to analyze this cylinder with Procedure SS8 were unsuccessful. Analytical results were simply not reasonable for this specimen. To determine if the problem was numerical in origin, Procedure SS8 was converted to double precision, but there was no change in the results. The problem is probably in the ring stiffener formulation, but no error could be found. Therefore, there are no analytical results for this specimen.

The natural frequencies and descriptions of the mode shapes determined experimentally are shown in Table XIII. The stiffened cylinder was tested with free-free boundary conditions. The technique used was the same as that used on the unstiffened cylinders.

SMD7076

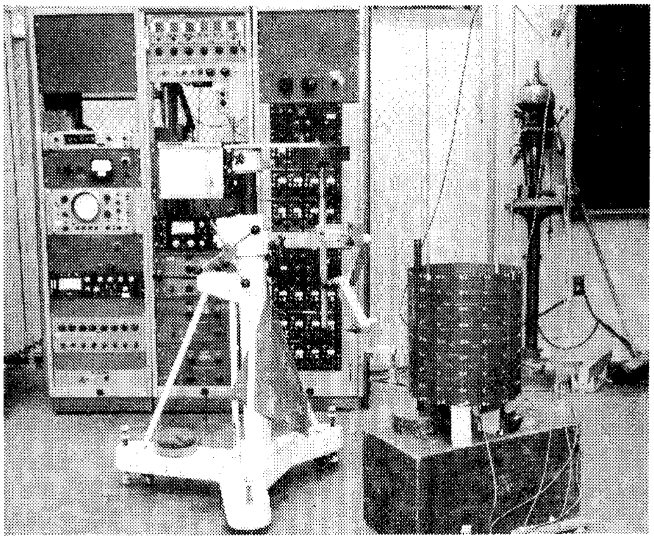


Figure 51 Dynamic Testing of a Cylinder

SMD7077

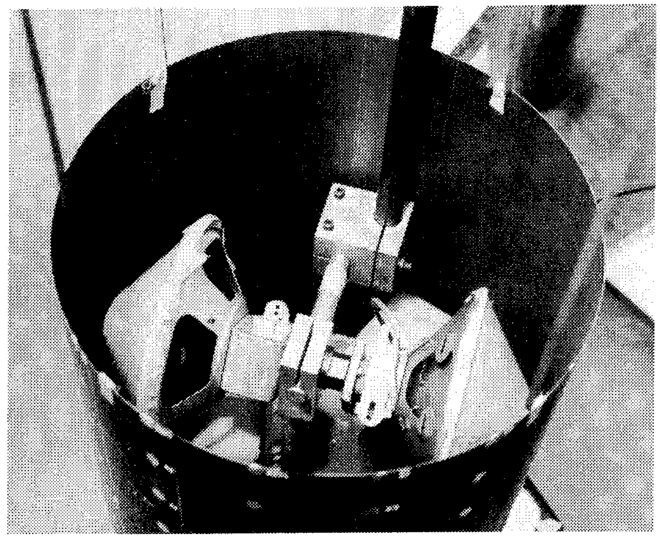


Figure 52 Dynamic Excitation of a Cylinder

SMD7078

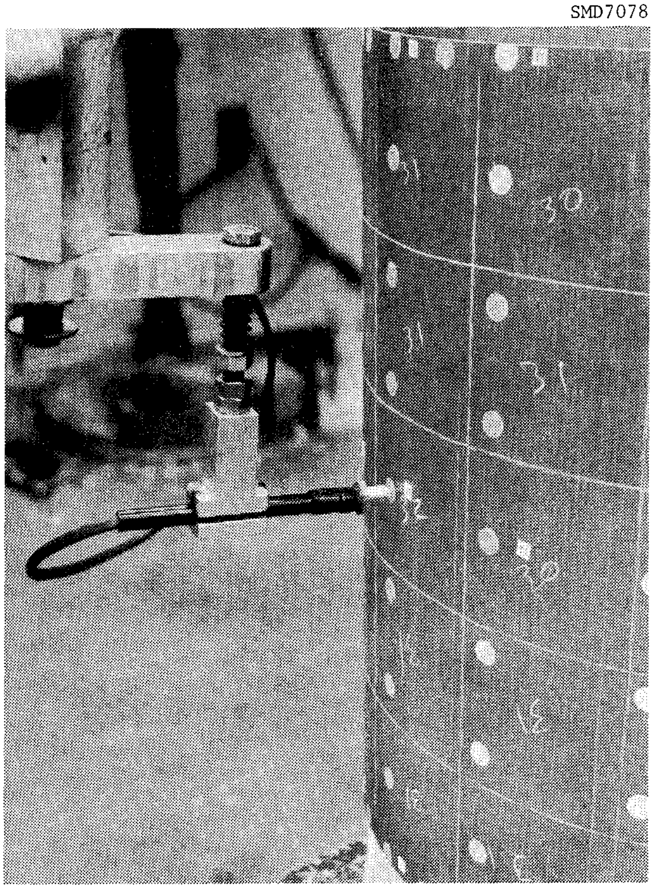


Figure 53 Modal Deflection Measurement

SMD7079

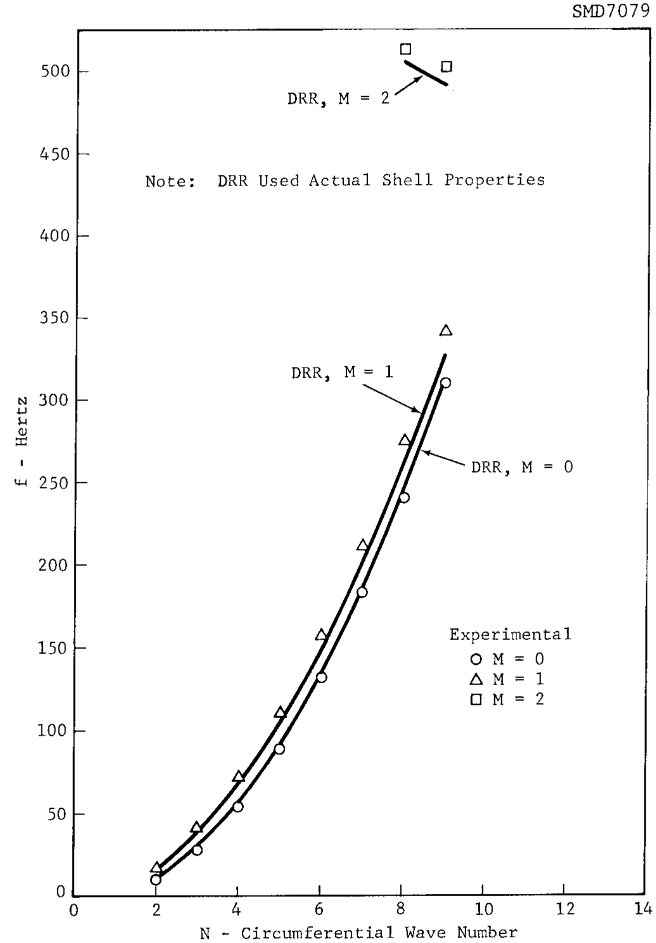


Figure 54 Frequency Correlation for Specimen 37

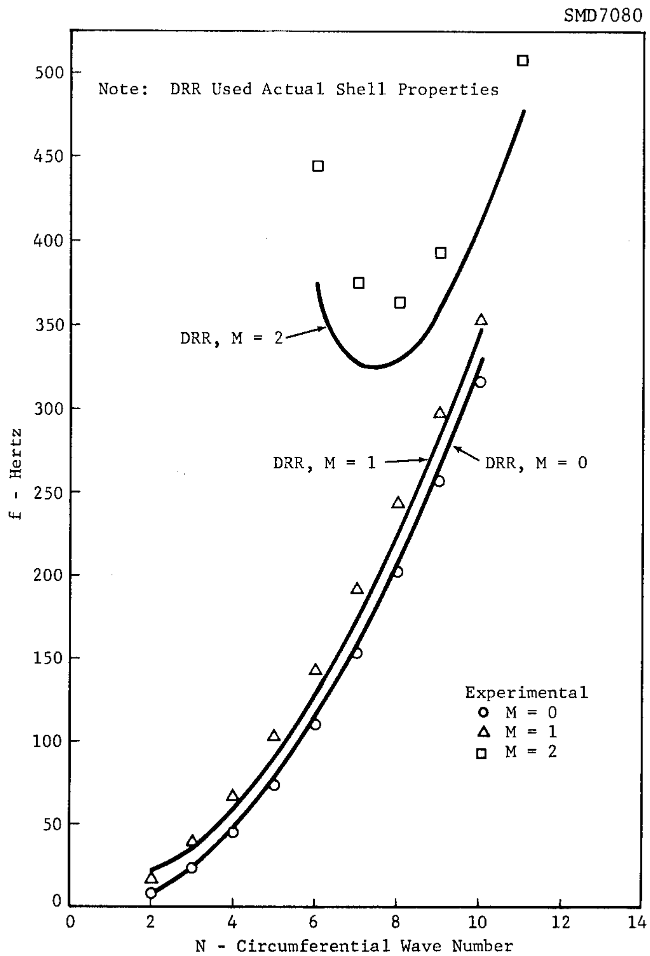


Figure 55 Frequency Correlation for Specimen 38

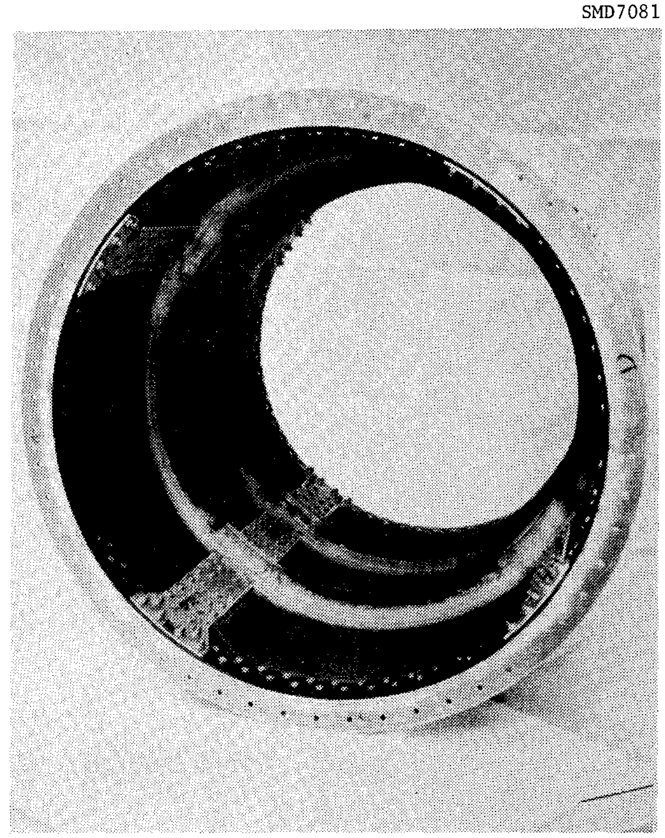


Figure 56 Graphite-Epoxy Stiffened Shell

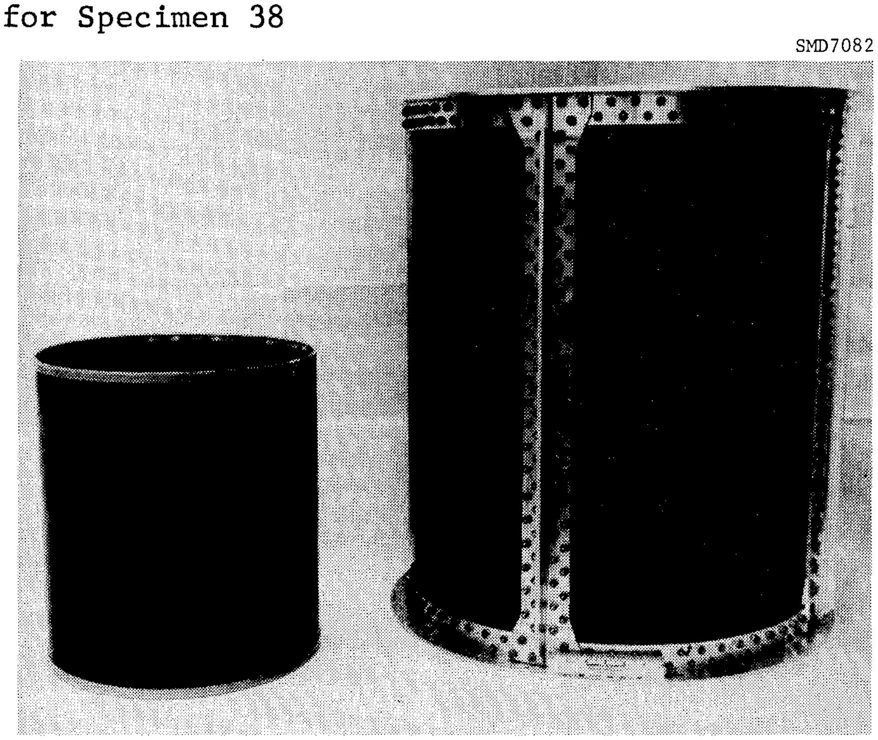


Figure 57 Stiffened and Unstiffened Cylinders

Table XI NATURAL FREQUENCIES (Hz) FOR UNSTIFFENED CYLINDERS

	SPECIMEN 37			SPECIMEN 38		
MODE	EXP	DRR	P.E.	EXP	DRR	P.E.
0,2	9.8	10.5	7.1	8.0	8.2	2.5
0,3	27.9	30.3	8.6	22.8	24.5	7.5
0,4	54.2	57.8	6.6	45.0	48.8	8.4
0,5	89.3	93.2	4.4	74.1	79.2	6.9
0,6	132	137	3.8	110	116	5.4
0,7	183	188	2.7	153	160	4.6
0,8	240	248	3.3	202	210	4.0
0,9	310	315	1.6	256	267	4.3
0,10	--	--	--	316	331	4.8
1,2	17.3	15.9	-8.1	16.5	22.7	37.6
1,3	41.5	38.5	-7.2	38.8	36.1	-7.0
1,4	72.9	68.0	-6.7	66.6	60.1	-9.8
1,5	111	104	-6.3	103	92.7	-10.0
1,6	157	148	-5.7	143	130	-9.1
1,7	211	200	-5.2	191	175	-8.4
1,8	274	260	-5.1	243	225	-7.4
1,9	341	327	-4.1	297	283	-4.7
1,10	--	--	--	353	347	-1.7
2,6	--	--	--	444	374	-15.8
2,7	--	--	--	375	327	-12.8
2,8	513	506	-1.4	363	329	-9.4
2,9	502	491	-2.2	392	362	-7.6
2,10	--	--	--	--	414	--
2,11	--	--	--	508	478	-5.9
Avg. P.E.	--	--	5.0	--	--	8.5

Table XII CYLINDER PROPERTIES

Property	Specimen 37		Specimen 38	
	Theory	Actual	Theory	Actual
W, lb.	1.648	1.678	1.099	1.089
t, in.	0.0312	0.03092	0.0208	0.0185
ℓ, in.	16.0	16.0	16.0	16.0
R, in.	7.5	7.5	7.5	7.5

Table XIII STIFFENED CYLINDER FREQUENCIES (Hz)

Frequency	Damping	Mode
129	.004	0,2 Nodes between stringers
152	.004	0,2 Nodes at stringers
362	.012	0,3
384	.008	1,3
498	.008	0,4
508	.110	2,3 Nodes at internal rings (1/3)
550	.018	2,2
582	.044	2,3 Nodes 20% from ends
589	.014	1,4
716	.070	2,2 & 6
735	--	-- 1st mode for center panels
933	.009	-- Not identifiable
967	.009	3,4
1275	.017	4,2

SECTION IV

SUMMARY

A Rayleigh-Ritz analysis for laminated anisotropic cylindrically curved shells has been performed. The analysis is formulated to solve static deflection, buckling, and natural vibration problems. Discrete energy contributions from stringers, rings, lumped masses, point loads, point and line moments, point and line springs, and elastic moment restraints have been included.

Digital computer Procedure SS8 has been written to compute the solutions to the above problems. The program has some limitations, mainly in regard to its treatment of free edges of a panel. The treatment of imperfection sensitivity in buckling should not be regarded as a final answer to the difficult problem of knockdown factors in compression, but did show promise. An assessment of the accuracy of the discrete ring stiffening capability was clouded by the problem of free edges. It is felt that the program serves a useful function as written, but that it needs more development work in certain areas.

Also described are various tests on curved panels and cylinders which in most cases were first attempts to discern the effects of curvature and anisotropy in laminated composites. Several interesting test methods were developed, including two applications of the Southwell method and an application of the Moire grid shadow technique.

REFERENCES

1. Ashton, J. E., Anisotropic Plate Analysis, General Dynamics Research and Development Report FZM-4899, October 1967.

2. Vlasov, V. Z., General Theory of Shells and Its Applications in Engineering, NASA Technical Translation, NASA-TTF-99, 1964.

3. Ashton, J. E.; Halpin, J. C.; and Petit, P. H., Primer on Composite Materials: Analysis, Technomic Publishing Company, Stamford, Connecticut, 1969.

4. Egle, D. M. and Soder, Jr., K. E., "A Theoretical Analysis of the Free Vibration of Discretely Stiffened Cylindrical Shells, With Arbitrary End Conditions," NASA CR-1316, 1969.

5. Ashton, J. E., "Analysis of Anisotropic Plates - II," J. Comp. Mat'l., 3, April 1969.

6. Ashton, J. E., Laminated Trapezoidal Plate Analysis, General Dynamics Research and Development Report FZM-5549, July 1970.

7. Love, T. S., and Brubaker, W. R., Advanced Composites - Buckling, Pressure and Natural Frequency Tests on Curved Panel Components, General Dynamics Test Laboratory Report FSG-739, December 1971.

8. Timoshenko, S. P. and Gere, J. M., Theory of Elastic Stability, Second Edition, McGraw-Hill, New York, 1961.

9. Dykes, B. C., "Analysis of Displacements in Large Plates by the Moire Grid-Shadow Technique, "Experimental Stress Analysis and its Influence on Design Conference, Cambridge England, April 1970, Proceedings, Institution of Mechanical Engineers, Publishers, London, 1970.

10. Tenerelli, D. J. and Horton, W. H., "An Experimental Study of the Local Buckling of Ring-Stiffened Cylinders Subject to Axial Compression, "Israel Journal of Technology, Vol. 7 No. 1-2, 1969, pp 181-194.

11. Galletly, G. D. and Reynolds, T. E., "A Simple Extension of Southwell's Method for Determining the Elastic General Instability Pressure,"Proceedings of the SESA, 13, (2), pp 141-152.

12. Horton, W. H. and Craig, J. I., "Experimental Studies of the Effect of General Imperfections on the Elastic Stability of Thin Shells," Israel Journal of Technology, 7, (1-2), 91.

13. McCullers, L.A., Naberhaus, J. D., and Bensinger, C. T., "Dynamic Characteristics of Advanced Filamentary Composite Structures, Volume I, Test Specimen Program, "AFFDL-TR-73-XX, Convair Aerospace, February 1973.

14. Sewall, J. L., Vibration Analysis of Cylindrically Curved Panels with Simply Supported or Clamped Edges and Comparison with Some Experiments, NASA TN D-3791, January 1967.

15. Park, A. C., et al., Dynamics of Shell-Like Lifting Bodies Part II. The Experimental Investigation, Technical Report AFFDL-TR-65-17, Part II, Wright-Patterson AFB, Ohio, June 1965.

16. Bert, C. W.; Baker, J. L.; and Egle, D. M., Free Vibrations of Multilayer Anisotropic Cylindrical Shells, presented to the Fourth Annual Symposium on High Performance Composites, Washington University, St. Louis, Missouri, 8 to 9 April 1969.

17. Schnell, W. and Heinrichsbauer, The Determination of Free Vibrations of Longitudinally Stiffened Thin-Walled, Circular Cylindrical Shells, NASA TTF-8856, April 1964.

18. Egle, D. M. and Soder, K. E., A Theoretical Analysis of the Free Vibration of Discretely Stiffened Cylindrical Shells with Arbitrary End Conditions, ASME/AIAA Tenth Structures, Dynamics, and Materials Conference, New Orleans,Louisiana, 1969.

19. Parthan, S. and Johns, D. J., Effects of Inplane and Rotary Inertia on the Frequencies of Eccentrically Stiffened Cylindrical Shells, ASME/AIAA Tenth Structures, Dynamics and Materials Conference, New Orleans, Louisiana, 1969.

20. Forsberg, K., "Influence of Boundary Conditions on the Modal Characteristics of Thin Cylindrical Shells," AIAA J., 2, 12, pp 2150-2156, December 1964.

21. Yamaki, N., "Flexural Vibrations of Thin Cylindrical Shells," 11th Structural Mechanics Conference, Japan Society of Mech. Engr., Kyoto, July 1969.

22. Cooper, P. A., "Effect of Shallow Meridional Curvature on the Vibration of Nearly Cylindrical Shells," NASA TN D-5143, April 1969.

23. Stavsky, Y and Loewy, R., "On the Vibrations of Heterogeneous Orthotropic Cylindrical Shells," Technion, Israel Inst. of Tech., AD709239, August 1969.

24. Kunukkasseril, V. X., "Vibration of Multi-Layered Anisotropic Cylindrical Shells," Watervliet Arsenal, AD649662, February 1967.

25. Dong, S. B., "Free Vibration of Laminated Orthotropic Cylindrical Shells," J. Acoustical Soc. of Amer., 44, 6, pp 1628-1635, 1968.

26. Tasi, J. and Roy, B. N., "Axisymmetric Vibrations of Finite Heterogeneous Cylindrical Shells," J. Sound Vib., 17, 1, pp 83-94, 1971.

27. Hsu, T. and Wang, J. T., "Rotationally Symmetrical Shells," J. Sound Vib., 16, 4, pp 473-487, 1971.

28. Stoneking, J. E., and Boresi, A. P., "A Theory for Free Vibration of Orthotropic Shells of Revolution," Nuclear Engr. & Design, 14, pp 271-285, 1970.

29. Sewall, J. L. and Naumann, E. C., "An Experimental and Analytical Vibration Study of Thin Cylindrical Shells with and Without Longitudinal Stiffeners," NASA TN D-4705, Sept. 1968.

30. Sewall, J. L., Thompson, Jr., W. G., and Pusey, C. G., "An Experimental and Analytical Vibration Study of Elliptical Cylindrical Shells," NASA TN D-6089, February 1971.

REFERENCES (CONT'D)

31. Sewall, J. L., Clary, R. R., and Leadbetter, S. A., "An Experimental and Analytical Vibration Study of a Ring-Stiffened Cylindrical Shell Structure With Various Support Conditions," NASA TN D-2398, August 1964.

32. Watkins, J. D. and Clary, R. R., "Vibrational Characteristics of Some Thin-Walled Cylindrical and Conical Frustum Shells," NASA TN D-2729, March 1965.

33. Maddox, N. R., Plumblee, H. E., and King, W. W., "Frequency Analysis of a Cylindrically Curved Panel With Clamped and Elastic Boundaries," J. Sound Vib., 12, 2, 225-249, 1970.

34. Gontkevich, V. S., Natural Vibrations of Plates and Shells, Naukova Dumka, Kiev, 1964, translated by Lockheed Missiles and Space Company.

35. Warburton, G. B., "Vibration of Thin Cylindrical Shells," J. Mech. Engr. Sci., 7, 4, 399-407, 1965.

36. Forsberg, K., "Axisymmetrical and Beam-Type Vibrations of Thin Cylindrical Shells," AIAA J. 7, 2, 221-227, February 1969.

37. Forsberg, K., "A Review of Analytical Methods Used To Determine the Modal Characteristics of Cylindrical Shells," NASA CR-613, Sept. 1966.

38. Nuckolls, C. E., and Egle, D. M., "The Influence of Elastic Boundary Conditions on the Resonant Response of Thin Cylindrical Shells," AIAA/ASME 12th Structures, Structural Dynamics, and Materials Conference, Anaheim, California, April 19-21, 1971.

39. Bray, F. M. and Egle, D. M., "An Experimental Investigation of The Free Vibration of Thin Cylindrical Shells with Discrete Longitudinal Stiffening," J. Sound Vib., 12, 2, 1970.

40. Basdekas, N. L., and Chi, M., "Response of Oddly-Stiffened Circular Cylindrical Shells," J. Sound Vib., 17, 2, 187-206, 1971.

REFERENCES (CONT'D)

41. McDonald, D., "A Problem in the Free Vibration of Stiffened Cylindrical Shells," AIAA J., 8, 2, 252-258, 1970.

42. Miller, P. R., "Free Vibration of a Stiffened Cylindrical Shell," Aeronautical Research Council L.A. 22, 19, 338, AD146359, 1957.

43. Miller, P. R., "Free Vibration of a Stiffened Cylindrical Shell," Aeronautical Research Council, R&M No. 3154, 1960.

44. Ligon, J. B., "Natural Frequencies of Stiffened Cylindrical Shells," Master's Thesis, University of Oklahoma, 1970.

45. Al-Najafi, A. M. J., and Warburton, G. B., "Free Vibration of Ring-Stiffened Cylindrical Shells," J. Sound Vib., 13, 1, 9-25, 1970.

46. Bert, C. W. and Egle, D. M., "Dynamics of Composite, Sandwich, and Stiffened Shell-Type Structures," J. Spacecraft & Rockets, 6, 12, 1345-1361, 1969.

47. Tsui, E. Y. W., "Natural Vibrations of Cylindrical Panels," J. Engr. Mech. Div., ASCE, 94, EM6, 1425-1445, 1968.

48. Reed, D. L., Point Stress Laminate Analysis, General Dynamics Research and Development Report FZM-5494, April 1970.

49. Tennyson, R. C., Chan, K. H., and Muggeridge, D. B., "The Effect of Axisymmetric Shape Imperfections on the Buckling of Laminated Anisotropic Circular Cylinders," Canadian Aeronautics and Space Institute, Transactions, 4, 2, September 1971.

50. Anon., "Buckling of Thin-Walled Circular Cylinders," NASA SP-8007, 1968.

51. Anon., System/360 Scientific Subroutine Package, H20-0205-3, IBM, 1968.

APPENDIX I

DESCRIPTION OF PROCEDURE SS8

The analysis described in Section II has been programed as IBM 370 Procedure SS8. Due to the large size of the program, a one-level, four-element overlay tree is used. The tree is shown in Figure 58. The longest resulting path is 418K bytes. All the subroutines are compiled under FORTRAN H, option 2, except subroutine ASEMBL, which is compiled with FORTRAN G.

Subroutines GSTART, PROB, SKIPPR, STATUS, and FREEFD are General Dynamics System Subroutines which perform I/O and timing functions. They would not be used elsewhere and are not discussed further. All other subroutines marked CF in Figure 58 are system-resident mathematical subroutines for matrix inversion or eigenvalue solutions. The purposes of the specially-written subroutines for SS8 are described below.

Main Program

The main program for SS8 serves only as a controller for implementing the necessary overlays. A blank common area and the labelled common blocks "CHECKS", "CNTROL", "NUMBER", "GEOM", "$TIME", "ABD", "PARAM", "VALUES", "ARRAYS", "BLOCK", "STFVAL", and "FLEXBL" are used for communication between overlays.

Subroutine READ

This subroutine reads all input data, based on the requirements of the problem, checks the input data, and does some preliminary calculations.

Subroutine CYLNDR

This subroutine calculates the appropriate running loads to be used when a force, torque, or bending moment is applied to a full cylinder. It should be noted that due to the uncoupling of

MAIN 'A'
BLANK (C.B.)
BLOCK (C.B.)
ARRAYS (C.B.)
VALUES (C.B.)
CNTROL (C.B.)
NUMBER (C.B.)
GEOM (C.B.)
$TIME (C.B.)
ABD (C.B.)
PARAM (C.B.)
CHECKS (C.B.)
STFVAL (C.B.)
FLEXBL (C.B.)
GSTART (CF)
PROB (CF)
SKIPPR (CF)
STATUS (CF)

- C.B. Denotes Common Block
- CF Denotes System Subroutines;
- ' ' Denotes Deck Identification Letter

READ 'B'	ASEMBL 'L'	SOLVE 'M'	PRINT 'S'
CHECK 'D'		EIGONE 'O'	DISPLA 'R'
CYLNDR 'C'		EIGALL 'P'	STRESS 'T'
STIFF 'E'		YOSFEM 'N'	ROTATE 'U'
TABLE 'F'		EGNVCT 'Q'	NRMLIZ 'V'
INTEGL 'G'		REDUCE 'X'	MARGIN 'W'
ELASTC 'H'		FLEX 'Y'	SEARCH 'K'
PPP 'I'		KDF 'Z'	OUT '/'
SPECAL 'J'		CUBIC '$'	
FREEFD (CF)		GJINV (CF)	
		QREIG (CF)	
		QRT (CF)	
		HESSEN (CF)	
		SIMEQ (CF)	
		COMMON (C.B.)	
		EIGWRK (C.B.)	
		EIGEN (C.B.)	
		ZWORK (C.B.)	
		CUBE (C.B.)	
		MIN ' '	

Figure 58 SS8 Overlay Structure

the axial and circumferential assumed mode shapes, torsional buckling results are not possible with SS8.

Subroutine CHECK

This subroutine writes a message and sets an error flag when subroutine READ detects an input error.

Subroutine STIFF

This subroutine calculates the A, B, and D stiffness terms as defined in Reference [3], and implemented in References [1] and [48], for a laminated plate.

Subroutine TABLE

This subroutine controls the calculation of the necessary integral tables of assumed modes in the x and y-directions.

Subroutine INTEGL

This subroutine, adapted from Reference [1], uses a highly efficient algorithm for calculating the necessary beam-mode integrals. By calling PPP and SPECAL, it calculates the single function integrals and the special cases for free-free and simple-free boundary conditions. At 625 points on the normalized shell surface, it calculates the value of the mode functions and their derivatives. At any stiffener locations, it calculates integrals, mode functions and derivatives.

Subroutine SPECAL

This subroutine calculates the integrals and mode constants for the simple-free and free-free boundary conditions.

Subroutine ELASTC

This subroutine implements the elastic moment restraint boundary condition by calculating the beam-mode constants which are dictated by the input moment restraint.

Subroutine PPP

This subroutine calculates the single-function beam-mode integrals.

Subroutine ASEMBL

Based on the input geometry and material properties and the calculated integrals, this subroutine assembles the matrices of potential energy, kinetic energy, lateral loads, and edge loads as required by the problem being performed. This assembly is done in submatrix fashion representing u, v, and w partitions.

Subroutine SOLVE

This subroutine uses the matrices from ASEMBL to solve the appropriate eigenvalue problem or simultaneous equations. It makes use of subroutines ARRAY, NROOT, and EIGEN from the IBM Scientific Subroutine Package.

Subroutine YOSFEM

This subroutine was written to perform multiplication of two large matrices by using a minimum amount of extra core storage. Optionally the product matrix may be stored in the premultiplier matrix or the postmultiplier matrix.

Subroutine EIGONE

For a single eigenvalue and eigenvector solution, the power method is an efficient algorithm. This method is used when a single buckling eigenvalue or frequency is desired.

Subroutine EIGALL

This subroutine finds all the eigenvalues of the matrix using the QR transform. The algorithm is programmed into three Convair Aerospace resident subroutines, HESSEN, QREIG, and QRT. Once the eigenvalues are found, the desired number of eigenvectors are found using a matrix decomposition technique in Subroutine EGNVCT.

Subroutine EGNVCT

Using the original matrix and a known eigenvalue, this routine uses matrix decomposition to find the corresponding eigenvector.

Subroutine PRINT

This subroutine performs various output functions, such as finding the dominant term in an eigenvector, calculating the problem execution time, and controlling other output subroutines.

Subroutine DISPLA

This subroutine calculates and prints deflections, curvatures, moments, shears, and edge reactions. All but edge reactions are printed at 625 equally-spaced points on the developed shell planform.

Subroutine OUT

This subroutine transforms the output arrays into a form for efficient printing.

Subroutine STRESS

This subroutine calculates stresses and strains at the 625 grid points.

Subroutine NRMLIZ

This subroutine finds the largest value in each output array and normalizes with respect to it.

Subroutine ROTATE

This subroutine performs a strain transformation of coordinates from one angle to another. It is used to check margins of safety in various directions.

Subroutine MARGIN

This subroutine calculates margins of safety according to the maximum strain theory of failure.

Subroutine SEARCH

This subroutine keeps track of the minimum margin of safety as well as its mode and location.

Subroutine FLEX

It is often desirable to determine an influence coefficient or flexibility matrix for a structure being analyzed. Since all of the problem types under consideration contain a term

$$[V] \quad \{a\}$$

where [V] is the varied strain energy density or the structural stiffness matrix in the generalized coordinates a_{imn}.

To obtain the point force-displacement flexibility matrix, the [V] matrix must first be partially inverted to produce the lateral stiffness matrix [S] in terms of the generalized lateral coordinates a_{3mn}. The stiffness matrix [S] may then be inverted and transformed from shape to point coordinates. The transformation matrix can be found from the expression for the lateral displacement at a point:

$$\delta_i = \sum_m \sum_n a_{3mn} X_{3m}(x_i) \, Y_{3n}(y_i)$$

where (x_i, y_i) are the coordinates of the i^{th} point. For N equations, this may be expressed in matrix form as

$$\{\delta_i\} = [R] \quad \{a\}$$

where [R] is the required transformation matrix. The desired flexibility matrix [F] can then be computed from

$$[F] = [R] \quad [S]^{-1} \quad [R]^T$$

at the N specified control points.

Subroutine REDUCE

This subroutine performs the partial inversion of the matrix containing membrane and bending degrees of freedom to reduce it to only its bending degrees of freedom.

Subroutine KDF

This subroutine uses the analysis of Reference [49] to account for imperfection sensitivity. It is an approximation since the Reference [49] analysis is done for a simply-supported full cylinder and relies on a precise definition of an axisymmetric imperfection. For the purpose of this study, the standard deviation of the thickness over the shell is used as a measure of imperfection, and the knockdown factor for the full cylinder is assumed to apply to any partial cylinder regardless of boundary conditions.

Subroutine CUBIC

This subroutine solves for the lowest real root of a cubic polynomial as required by KDF. This is done by Newton-Raphson iteration for the first root, and then by synthetic division and the quadratic formula for the other two.

Subroutine MIN

This is a general subroutine for determining the smallest element in a vector of values.

Subroutine SWITCH

This subroutine is used in the matrix operations of subroutine SOLVE. It changes diagonal elements in a matrix from 0. to 1. or vice-versa. It is used to prevent the singular matrices (which arise for some problems involving rigid-body modes) from inhibiting a solution.

APPENDIX II

CUSTOMER INSTRUCTIONS FOR SS8

PROCEDURE SS8

Anisotropic Curved Panel Analysis Program

21 January 1970
D. J. Wilkins

PROBLEM DESCRIPTION

This procedure analyzes cylindrically curved panels with respect to dynamic response, buckling, and static deflection. Vlasov shell theory is used for the formulation and the Rayleigh-Ritz energy method is used for the solution. The integral generation scheme from Procedure RA5 is also employed.

The procedure is capable of analyzing flat plates, cylindrically curved panels, and full cylinders. All combinations of clamped, and simply supported edges, and some combinations of free edges may be specified. Elastic boundary restraint may also be specified.

The material may be isotropic, a laminate of identical orthotropic layers, a laminate of dissimilar orthotropic layers, or a sandwich with orthotropic facings. (No transverse shear effects are included, so that the sandwich analysis is only appropriate for stiff cores.) Discrete, eccentric rings and stringers may be specified.

Edge loads and lateral loads may be specified by up to tenth order polynomials. Point loads, point moments, and line moments may also be used, as well as point and line spring supports. In dynamics, the effects of lumped masses may be included.

In any one problem, the procedure can solve for natural frequencies and mode shapes, or the buckling stress resultants under complicated edge load distributions, or the static deflections (including stresses, strains, and margins of safety) under lateral and edge loads. A flexibility matrix at specified control points may be calculated on any type problem.

INPUT DATA

The program uses "free field" input as explained in the documentation for general purpose subroutine CF619. However, every number input as problem data is considered by the program to be a real number (card type "6" in free field). Therefore, every card of the input deck should have a "6" in column 1. It should be noted that if an input number is an integer, a decimal point is not necessary. The title card (Card No. 1) is not read in the free field mode but it also contains a "6" in column 1.

The general content of each card in a problem deck is as follows:

Column	
1	The integer "6"
2 -66	Input data
67 - 72	The six-digit job number
73	The letter "P"
74 - 75	The problem number, beginning with 01
76 - 79	The card sequence number, beginning with 0001.

The input data varies according to the problem being run. A flow chart of the necessary data to run a given problem is shown in Figure 59. One or more cards may be required for each block of data, but each block must begin on a new card.

A description of the data blocks follows:

Block 1. Title

Printed with the output. Any Fortran characters may be used. (1 card only.)

Block 2. IFLAGD, IFLAGB, IFLAGW, IBCX, IBCY, NTX, NTY, ITX, ITY, NMODES, IMATL, NPLYS, IREACT, IOUT, IEDGE, NPNX, NPNY, IPRTN, NQTX, NQTY, IPRTQ, NSTRNG, NRING, NLMASS, NPTLDS, NPTMOM, NLNMOM, NPTSUP, NLNSPR, INTPRT, IFLEX (31 integers)

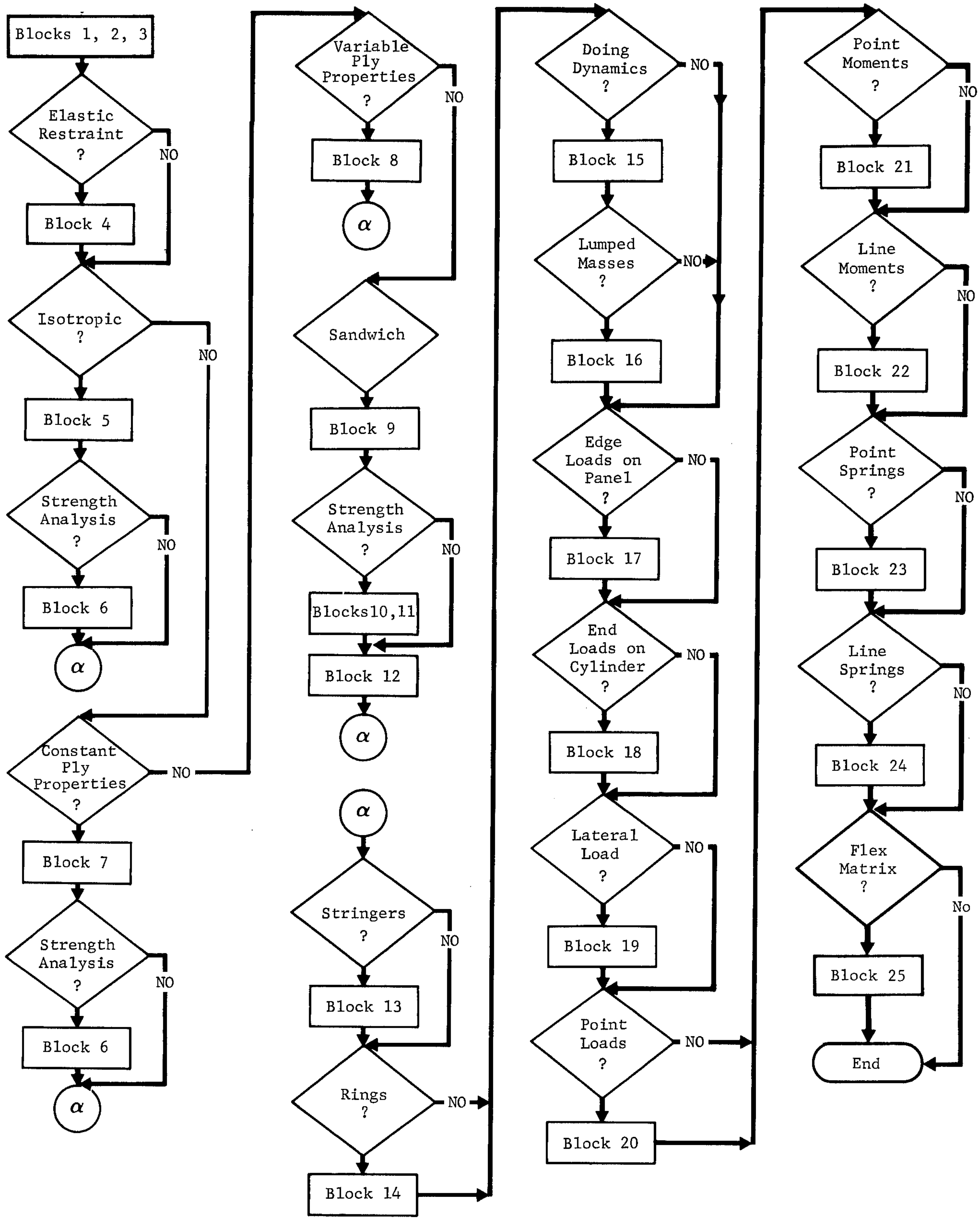


Figure 59 Input Data Flow Chart

IFLAGD = +1, if doing a dynamics problem
= +0, otherwise.

IFLAGB = +1, if 1 buckling eigenvalue is desired
= +2, if 2 buckling values are desired (as for shear buckling)
= +3, if 1 buckling eigenvalue and an imperfection sensitivity analysis are desired
= +4, if 2 buckling eigenvalues and an imperfection sensitivity analysis are desired
= +0, otherwise.

IFLAGW = +1, if doing a deflection analysis with lateral pressure, q
= +2, if doing a deflection analysis with no lateral pressure, q
= +0, otherwise.

IBCX is a tag for the boundary condition in the x-direction.
= +1, for clamped-simply supported
= +2, for simply supported-simply supported
= +3, for clamped-clamped
= +4, for clamped-free
= +5, for simply supported-free
= +6, for free-free
= +7, for elastic restraint. ($w_{,xx} = \alpha_x w_{,x}\big|_{x=0}$,
$w_{,xx} = \beta_x w_{,x}\big|_{x=a}$)

IBCY is a tag for the boundary condition in the y-direction.
= +0, for a full cylinder
= +1, for clamped-simply supported
= +2, for simply supported-simply supported
= +3, for clamped-clamped
= +4, for clamped-free
= +5, for simply supported-free
= +6, for free-free
= +7, for elastic restraint exactly the same as that in the x-direction
= +8, other elastic restraint. ($w_{,yy} = \alpha_y w_{,y}\big|_{y=0}$
$w_{,yy} = \beta_y w_{,y}\big|_{y=b}$)

NTX = Number of terms in the assumed series for u, v, and w, in the x-direction. $1 \leq NTX \leq 10$.

NTY = Number of terms in the assumed series for u, v, and w, in the y-direction. $1 \leq NTY \leq 10$.

Note: Although the upper limit on each of the above two numbers is ten, the limit on the size of the matrices generated using them is 150. This means that NTX * NTY ≤ 50.

ITX = The beginning term in the assumed series for u, v, and w. This number sets the range of m (axial wave number) to be considered in the analysis, such that $ITX \leq M \leq ITX + NTX - 1$. The range on ITX is $1 \leq ITX \leq 20$.

ITY = The beginning term in the assumed series for u, v, and w. This number sets the range of n (circumferential wave number) to be considered in the analysis, such that $ITY \leq N \leq ITY + NTY - 1$. The range on ITY is $1 \leq ITY \leq 20$.

NMODES = Number of mode shapes to be calculated in a natural frequency problem. $1 \leq NMODES \leq 20$.
= +0, for a buckling or lateral loads problem.

IMATL = +1, for an isotropic material
= +2, for a laminate with constant ply properties
= +3, for a laminate with variable ply properties
= +4, for a sandwich with orthotropic facings.

NPLYS = Number of plys in the laminate $1 \leq NPLYS \leq 40$. For an isotropic material, NPLYS = +1. For a sandwich, NPLYS = +3.

IREACT = +1 if the reactions (at the corners, or along the edges of the panel, or at elastic supports) are desired.
= +0, otherwise.

IOUT = An indicator that controls how much output is given and also controls whether a lamina strength analysis is performed. Each of the following output quantities is printed at 625 points over the panel, with the x = 0 axis across the top and the y = 0 axis down the left hand side.

= +1, for printing the normal deflection, w, only
= +2, for printing w, u, and v (mid-surface deflections)
= +3, for printing w, u, v, $\epsilon_x^o, \epsilon_y^o, \epsilon_{xy}^o$ (mid-surface strains) and K_x, K_y, K_{xy} (curvatures)
= +4, for printing w, u, v, M_x, M_y, M_{xy} (moment resultants), Q_x, Q_y, (transverse shear resultants), and σ_x, σ_y, σ_{xy} (stresses, only for isotropic or sandwich)
= +5, for printing w, u, v, M_x, M_y, Q_x, Q_y, ϵ_x^o, ϵ_y^o, ϵ_{xy}^o, K_x, K_y, K_{xy}, σ_x, σ_y, σ_{xy}
= +6, for printing w, σ_x, σ_y, σ_{xy}
= +7, for printing w, ϵ_1, ϵ_2, ϵ_{12} (strains in lamina axes for each ply), $M.S._1$, $M.S._2$, $M.S._{12}$ (margins of safety for each ply according to the maximum strain theory)
= +8, for printing w, σ_x, σ_y, σ_{xy}, ϵ_1, ϵ_2, ϵ_{12}, $M.S._1$, $M.S._2$, $M.S._{12}$
= +9, for printing w, u, v, M_x, M_y, M_{xy}, Q_x, Q_y, ϵ_x^o, ϵ_y^o, ϵ_{xy}^o, K_x, K_y, K_{xy}, σ_x, σ_y, σ_{xy}, ϵ_1, ϵ_2, ϵ_{12}, $M.S._1$, $M.S._2$, $M.S._{12}$.

IEDGE = +1, if edge loads are to be input
= +2, if cylinder end loads (force, torque, bending moment are to be input)
= +0, otherwise.

NPNX = Number of terms in the edge loads expressions in the x-direction. $1 \leq NPNX \leq 10$.
= +0, if IEDGE = +0 or +2.

NPNY = Number of terms in the edge loads expressions in the y-direction. $1 \leq NPNX \leq 10$.
= +0, if IEDGE = +0 or +2.

IPRTN = +1, if the distributions of the edge loads are to be printed at quarter points of the panel.
= +0, otherwise.

NQTX = Number of terms in the distributed lateral loads expression in the x-direction. $1 \leq NQTX \leq 10$.
= +0, if IFLAGW = +0 or +2.

NQTY = Number of terms in the distributed lateral loads expression in the y-direction. $1 \leq NQTY \leq 10$.
= +0, if IFLAGW = +0 or +2.

IPRTQ = +1, if the distribution of the lateral loads is to be printed at quarter points of the panel.
= +0, otherwise.

NSTRNG = Number of stringers. $0 \leq NSTRNG \leq 100$. (For equally-spaced identical stringers, precede number by a minus sign.)

NRING = Number of rings. $0 \leq NRING \leq 50$. (For equally-spaced identical rings, precede number by a minus sign.)

NLMASS = Number of lumped masses. $0 \leq NLMASS \leq 50$.

NPTLDS =Number of concentrated normal loads. $0 \leq NPTLDS \leq 50$.

NPTMOM = Number of concentrated point moments. $0 \leq NPTMOM \leq 50$.

NLNMOM = Number of concentrated line moments. $0 \leq NLNMOM \leq 50$.

NPTSUP = Number of point spring supports. $0 \leq NPTSUP \leq 50$.

NLNSPR = Number of line spring supports. $0 \leq NLNSPR \leq 50$.

INTPRT = +1, if the values of the calculated integrals, the matrices generated, and detailed timing information is to be printed.
= +0, otherwise. (Usually, INTPRT = +0).

IFLEX = Number of points for which influence coefficients are desired.

Block 3. AA, [BB], RR, [MU]

AA = Dimension in the x-direction

BB = Dimension in the y-direction (Note: This is not input for a full cylinder.)

RR = Radius of panel.

MU = Standard deviation of panel thickness.

Block 4. [ALFAX, BETAX], [ALFAY, BETAY]

ALFAX = The constant describing the elastic restraint on the edge $x = 0$. $w_{,xx} = (\text{ALFAX})w_{,x}$.

BETAX = The constant describing the elastic restraint on the edge $x = a$. $w_{,xx} = (-\text{BETAX})w_{,x}$.

ALFAY = The constant describing the elastic restraint on the edge $y = 0$. $w_{,yy} = (\text{ALFAY})w_{,y}$.

BETAY = The constant describing the elastic restraint on the edge $y = b$. $w_{,yy} = (-\text{BETAY})w_{,y}$.

The elastic restraint constants are only input as needed, and if the y-direction quantities are identical to those in the x-direction, only ALFAX and BETAX need be input. All of these constants are input as positive for positive restraint.

Block 5. E, ν, T

E = Young's modulus, psi

ν = Poisson's ratio, dimensionless

T = Panel thickness, in.

Block 6. EC (1), EC(2), EC(3), ET(1), ET(2), ET(3)

EC(1) = Compressive strain allowable in the 1-direction, in/in.

EC(2) = Compressive strain allowable in the 2-direction, in/in.

EC(3) = Negative shear strain allowable, in/in.

ET(1) = Tensile strain allowable in the 1-direction, in/in.

ET(2) = Tensile strain allowable in the 2-direction, in/in.

ET(3) = Positive shear strain allowable, in/in.

Block 7. E1, E2, G, ν_{12}, H, (θ_i, i = 1,2,....,NPLYS)

E1 = Modulus in the 0^o direction, psi.

E2 = Modulus in the 90^o direction, psi.

G = In-plane shear modulus, psi.

ν_{12} = Major Poisson's ratio, dimensionless.

H = Thickness of each ply, in.

θ_i = Orientation of the i^{th} ply, starting with the bottom or inner ply, degrees.

Block 8. $(E1)_i$, $(E2)_i$, G_i, $(\nu_{12})_i$, H_i, θ_i, [$EC(1)_i$, $EC(2)_i$, $EC(3)_i$, $ET(1)_i$, $ET(2)_i$, $ET(3)_i$], i = 1,...., NPLYS

$E1_i$ = Modulus in the 0^o direction of the i^{th} ply,psi

$E2_i$ = Modulus in the 90^o direction of the i^{th} ply, psi

G_i = Shear modulus of the i^{th} ply, psi

$(\nu_{12})_i$ = Major Poisson's ratio of the i^{th} ply, dimensionless

H_i = Thickness of the i^{th} ply, in.

θ_i = Orientation of the i^{th} ply, degrees.

(The following allowables are input only if a strength analysis is being performed, IOUT $\geq$ 7.)

$EC(1)_i$ = Compressive strain allowable in the 1-direction for the i^{th} ply, in/in.

$EC(2)_i$ = Compressive strain allowable in the 2-direction for the i^{th} ply, in./in.

$EC(3)_i$ = Negative shear strain allowable in the 1-2 plane for the i^{th} ply, in/in.

$ET(1)_i$ = Tensile strain allowable in the 1-direction for the i^{th} ply, in/in.

$ET(2)_i$ = Tensile strain allowable in the 2-direction for the i^{th} ply, in/in.

$ET(3)_i$ = Positive shear strain allowable in the 1-2 plane for the i^{th} ply, in/in.

Block 9. E1, E2, G, ν_{12}, H

E1 = Inner (outer) facing modulus in the 0^{o} direction, psi.

E2 = Inner (outer) facing modulus in the 90^{o} direction, psi.

G = Inner (outer) facing shear modulus, psi.

ν_{12} = Inner (outer) facing major Poisson's ratio, dimensionless.

H = Inner (outer) facing thickness, in.

(If a strength analysis <u>is not</u> being performed, Block 9 is now repeated for the outer facing properties. If a strength analysis <u>is</u> being performed, Blocks 10 and 11 for the inner facing are now input, then Blocks 9, 10 and 11 are input for the outer facing.)

Block 10. EC(1), EC(2), EC(3), ET(1), ET(2), ET(3), MCHK

EC(1) = Inner (outer) facing compressive strain allowable in the 1-direction, in/in.

EC(2) = Inner (outer) facing compressive strain allowable in the 2-direction, in/in.

EC(3) = Inner (outer) facing negative shear strain allowable in the 1-2 plane, in/in.

ET(1) = Inner (outer) facing tensile strain allowable in the 1-direction, in/in.

ET(2) = Inner (outer) facing tensile strain allowable in the 2-direction, in/in.

ET(3) = Inner (outer) facing positive shear strain allowable in the 1-2 plane, in/in.

MCHK = Number of orientations to be checked in the strength analysis of the inner (outer) facing. $1 \leq \text{MCHK} \leq 10$.

Block 11. ANGCHK_i, i = 1, MCHK

ANGCHK_i = Orientations to be checked in the strength analysis of the inner (outer) facing, degrees.

Block 12. H_c

H_c = Core thickness, in.

Block 13. [YSTRNG], YBAR, ZBAR, AS,XIYYS, XIYZS, XIZZS, ES, GJS, RHOS

YSTRNG = Distance of longitudinal stiffener from y = 0. For variable stiffener spacing only.

YBAR = Location of stringer centroid in the y-direction with respect to its line of attachment to the shell, in.

ZBAR = Location of stringer centroid in the z-direction with respect to the middle surface of the shell at the line of attachment, in.

AS = Stringer cross-sectional area, in^2.

XIYYX = Moment of inertia of the stringer area about the mid-surface y- axis at the line of attachment, in^4.

XIYZS = Product of inertia of the stringer area about the mid-surface y-z axis at the line of attachment, in^4.

XIZZS = Moment of inertia of the stringer area about the z-axis at the line of attachment, in^4.

ES = Stringer modulus of elasticity,psi.

GJS = Stringer torsional stiffness, lb-in.^{2}.

RHOS = Average density of stringer material, $lb\text{-}sec^2/in^4$.

Block 13 is repeated 'NSTRNG' times, unless equally-spaced identical stringers were specified.

Block 14. [XRING], XBARR, ZBARR, AR, XIXXR, XIZZR, ER, GJR, RHOR

XRING = Distance of circumferential stiffener from x = 0. For unequally spaced rings.

XBARR = Location of ring centroid in the x-direction with respect to its line of attachment to the shell, in.

ZBARR = Location of ring centroid in the z-direction with respect to the middle surface of the shell at the line of attachment, in.

AR = Ring cross-sectional area, in^2.

XIXXR = Moment of inertia of the ring area about the mid-surface x-axis at the line of attachment, $in.^4$.

XIXZR = Product of inertia of the ring area about the mid-surface x-z axis at the line of attachment, in^4.

XIZZR = Moment of inertia of the ring area about the z-axis at the line of attachment, in^4.

ER = Ring modulus of elasticity, psi.

GJR = Ring torsional stiffness, $lb\text{-}in^2$.

RHOR = Average density of ring material, $lb\text{-}sec^2/in^4$.

Block 14 is repeated 'NRING' time unless equally-spaced identical rings were specified.

Block 15. DENSE

DENSE = Average material density of the shell material, such that (DENSE) (Vol. of shell) = (Mass of shell), $lb\text{-}sec^2/in^4$.

Block 16. IX, IY, PMASS

IX = Grid coordinate in x-direction at which lumped mass is located, $1 \leq IX \leq 25$.

IY = Grid coordinate in y-direction at which lumped mass is located, $1 \leq IY \leq 25$.

PMASS = Mass, lb-sec^2/in.

Block 16 is repeated 'NLMASS' times.

Block 17. PX(1,1), PY(1,1), PXY(1,1), PX(2,1), PY(2,1), PXY(2,1), ...PX(I,J), PY(I,J), PXY(I,J), I = 1,2...NPNX, J = 1,2 ...NPNY

The applied in-plane stress resultants are described by the relations

$$N_x(x,y) = \sum_{I=1}^{NPNX} \sum_{J=1}^{NPNY} P_x(I,J) \left(\frac{x}{a}\right)^{I-1} \left(\frac{y}{b}\right)^{J-1}$$

$$N_y(x,y) = \sum_{I=1}^{NPNX} \sum_{J=1}^{NPNY} P_y(I,J) \left(\frac{x}{a}\right)^{I-1} \left(\frac{y}{b}\right)^{J-1}$$

$$N_{xy}(x,y) = \sum_{I=1}^{NPNX} \sum_{J=1}^{NPNY} P_{xy}(I,J) \left(\frac{x}{a}\right)^{I-1} \left(\frac{y}{b}\right)^{J-1}$$

Note: Tension stress resultant are taken as positive.

TORQUE = Torque applied to cylinder, in-lb.

BNDMOM = Bending moment applied to cylinder, in-lb.

Block 19. Q(1,1), Q(2,1), Q(3,1), ...Q(I,J), I = 1, ..., NQTX J = 1,2..., NQTY

The distributed lateral load is described by the relation

$$q(x,y) = \sum_{I=1}^{NQTX} \sum_{J=1}^{NQTY} Q(I,J) \left(\frac{x}{a}\right)^{I-1} \left(\frac{y}{b}\right)^{J-1}$$

Note: positive loads are in the positive z-direction.

Block 20. IX, IY, PC

IX = Grid coordinate in x-direction, $1 \leq IX \leq 25$.

IY = Grid coordinate in y-direction, $1 \leq IY \leq 25$.

PC = Concentrated load, lb.

Block 20 is repeated 'NPTLDS' times.

Block 21. IX, IY, ITAG, FC

IX = Grid coordinate in x-direction, $1 \leq IX \leq 25$.

IY = Grid coordinate in the y-direction, $1 \leq IY \leq 25$.

ITAG = +1, if the moment is about the x-axis in a vector sense (right-hand rule)
= +2, if the moment is about the y-axis.

FC = Moment, in-lb.

Block 21 is repeated 'NPTMOM' times.

Block 22. ITAG, IDIST, PLMOM

ITAG = +1, if the line moment is parallel to the x-axis.
= +2, if the line moment is parallel to the y-axis.

IDIST = Number of grid lines away from the x = o or y = o axis. $1 \leq IDIST \leq 25$.

PLMOM = Line moment per unit of length, in-lb/in.

Block 22 is repeated 'NLNMOM' times.

Block 23. IX, IY, PKC

IX = Grid coordinate in x-direction. $1 \leq IX \leq 25$.

IY = Grid coordinate in y-direction. $1 \leq IY \leq 25$.

PKC = Spring constant, lb/in.

Block 23 is repeated 'NPTSUP' times.

Block 24. ITAG, IDIST,PLINE

ITAG = +1, if the line spring is parallel to the x-axis.
= +2, if the line spring is parallel to the y-axis.

IDIST = Number of grid lines away from the x=o or y=o axis. $1 \leq \text{IDIST} \leq 25$.

PLINE = Spring constant per unit length, lb/in^2.

Block 24 is repeated 'NLNSPR' times.

Block 25. XP(I), YP(I), I = 1, IFLEX

XP(I) = X-coordinate (in %) of I^{th} flexibility matrix control point.

YP(I) = Y-coordinate (in %) of I^{th} flexibility matrix control point.

OUTPUT DATA DESCRIPTION

Most of the output is labeled with the exception of the 'CONTROBUTIONS OF THE SERIES TERMS'. These are the solution vectors used for the modal analysis. They are printed in the following order:

$$a_{111}, a_{112}, a_{113}, \ldots, a_{11(NTY)}, a_{121}, a_{122}, \ldots, a_{12(NTY)},$$
$$\ldots, a_{1(NTX)1}, a_{1(NTX)2}, \ldots, a_{1(NTX)NTY)}, a_{211}, a_{212},$$
$$\ldots, a_{2(NTX)(NTY)}, a_{311}, a_{312}, \ldots, a_{3(NTX)(NTY)}$$

where

$$u = \sum_{m=M_i}^{M_f} \sum_{n=N_i}^{N_f} a_{1mn} X_{1m} Y_{1n}$$

$$v = \sum_{m=M_i}^{M_f} \sum_{n=N_i}^{N_f} a_{2mn} X_{2m} Y_{2n}$$

$$w = \sum_{m=M_i}^{M_f} \sum_{n=N_i}^{N_f} a_{3mn} X_{3m} Y_{3n}$$

M_i = ITX.

M_f = ITX + NTX -1.

N_i = ITY.

N_f = ITY + NTY -1.

For a buckling solution only the a_{3mn} are printed.

RESTRICTIONS

The ranges of the input parameters are described under INPUT DATA.

The main restriction is to keep in mind the assumptions of the analysis, particularly the small-deflection assumption. If the deflections found in a lateral loads problem are greater than the panel thickness, the results are questionable.

If a solution mode shape contains large contributions from the highest modal shape input, the solution is questionable, and the analysis should be rerun using the highest mode shape input as the initial term in the new analysis. Since the high-order modes are not sensitive to boundary conditions, the restriction to simply-supported or full cylinder boundary conditions will not make much difference in the results.

ESTIMATED RUNNING TIME

The run times may vary considerably depending solely on the size of the matrix to be inverted and solved for eigenvalues. A meaningful buckling problem may be solved in 10 to 20 seconds, while a large vibration problem with many mode shapes desired may run up to 10 minutes. For the static deflection and buckling problem, an estimate of the run time can be obtained as

$$t = 9.4^{0.0666\,(NTX*NTY)} \text{ sec.}$$

The vibration problems normally run up to twice as long as the corresponding buckling problems, and can run longer when many modes are desired.

A P P E N D I X I I I

S A M P L E P R O B L E M S

```
6     59A                                                      004602P540001
6 +++1 +3+3 +5+5 +1+1 ++2+6 ++1+++++1+1+++++++++++            004602P540002
6 +12 +8 +12 +.0010                                           004602P540003
6+21000000+1700000+650000+.21+.0070+-60+60+60-60+              004602P540004
6 +1++                                                         004602P540005
```

59A

THE BOUNDARY CONDITIONS AT X=0 AND X=A ARE
CLAMPED, CLAMPED

THE BOUNDARY CONDITIONS AT Y=0 AND Y=B ARE
CLAMPED, CLAMPED

THERE ARE 5 MODES IN THE X DIRECTION, STARTING WITH M = 1 .
THERE ARE 5 MODES IN THE Y DIRECTION, STARTING WITH N = 1 .

THE STIFFNESS MATRIX SIZE IS 75 BY 75

A SOLUTION UNDER LATERAL LOADS WILL BE SOUGHT

A = 12.00000

B = 8.00000

R = 12.00000

MU = 0.0

FOR THE 6 PLY LAMINATE

E1 = 0.210000E 08

E2 = 0.170000E 07

G = 0.650000E 06

NU12 = 0.2100

H(I) = 0.0070

T = 0.0420

THE ORIENTATIONS ARE
0.0

-60.0000

60.0000

60.0000

-60.0000

0.0

THE CONSTITUTIVE MATRIX IS

0.3762170E 06	0.1172375E 06	0.0	0.3906250E-02	0.2136230E-03	0.0
0.1172375E 06	0.3762169E 06	0.0	0.2136230E-03	0.0	0.0
0.0	0.0	0.1294898E 06	0.0	0.0	0.1678467E-03
0.3906250E-02	0.2136230E-03	0.0	0.9686734E 02	0.8888407E 01	-0.2862569E 01
0.2136230E-03	0.0	0.0	0.8888407E 01	0.3043127E 02	-0.8644495E 01
0.0	0.0	0.1678467E-03	-0.2862569E 01	-0.8644495E 01	0.1068948E 02

THE LAMINATE PROPERTIES ARE

EX = 0.808769E 07 EY = 0.808769E 07 G = 0.308309E 07 NUXY = 0.3116 NUYX = 0.3116

Q(I,J) FOLLOWS
1.0000E 00

THE CONTRIBUTIONS OF THE SERIES TERMS TO DEFLECTION FOLLOW

0.5184E-06	-0.3113E-16	0.3784E-06	0.4528E-16	0.1862E-06	0.1995E-16	0.9983E-09	-0.2213E-17	-0.1041E-08	-0.3966E-19
0.1313E-06	-0.3206E-17	0.3767E-07	0.2823E-17	0.3791E-07	0.5546E-17	-0.8079E-09	0.6264E-18	0.9691E-09	0.1576E-17
0.3827E-07	-0.1220E-17	0.1065E-07	-0.8611E-18	0.8614E-08	0.4368E-05	0.4131E-16	0.1860E-06	-0.4679E-16	-0.1867E-07
-0.6900E-16	-0.4840E-08	-0.4833E-17	0.8495E-09	0.1098E-17	0.1061E-05	0.1504E-17	0.1792E-06	-0.6508E-18	0.6608E-08
0.2420E-16	0.6866E-08	0.4198E-17	-0.4262E-08	0.3072E-19	0.4348E-06	0.4974E-17	0.8720E-07	0.5054E-17	0.1355E-07
0.3430E-03	0.3016E-14	0.7821E-04	-0.1087E-13	0.8165E-05	-0.5216E-14	-0.5493E-06	-0.1595E-14	0.2288E-06	0.3261E-15
0.1310E-03	-0.6141E-15	0.5687E-04	0.9607E-15	0.1286E-04	0.4071E-14	0.4955E-06	0.1628E-14	-0.1509E-05	0.8037E-15
0.6974E-04	-0.3629E-15	0.3075E-04	0.1811E-14	0.1175E-04					

THE W DEFLECTIONS DIVIDED BY 0.622055E-03/10000 FOLLOW

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	152	495	872	1168	1330	1367	1326	1264	1222	1209	1212	1212	1201	1190	1201	1248	1319	1370	1342	1184	887	505	156	0
0	469	1527	2696	3625	4151	4290	4185	4006	3876	3831	3834	3830	3800	3773	3813	3957	4163	4301	4189	3676	2743	1558	479	0
0	764	2494	4423	5984	6903	7197	7077	6815	6607	6520	6509	6498	6457	6432	6511	6741	7047	7217	6965	6066	4498	2542	780	0
0	914	3002	5363	7325	8549	9028	8987	8730	8491	8364	8324	8301	8269	8272	8391	8655	8960	9055	8622	7418	5447	3056	932	0
0	889	2945	5324	7381	8769	9440	9568	9415	9204	9053	8976	8943	8938	8990	9138	9368	9559	9473	8834	7460	5393	2989	903	0
0	739	2485	4576	6492	7919	8763	9106	9122	8992	8843	8740	8702	8734	8835	8988	9129	9126	8796	7961	6534	4610	2505	746	0
0	561	1926	3643	5333	6731	7707	8253	8451	8428	8312	8208	9178	8241	8371	8498	8519	8306	7738	6741	5327	3631	1916	557	0
0	442	1553	3014	4541	5908	6966	7654	7996	8080	8026	7953	7943	8018	8139	8211	8113	7731	6992	5888	4493	2962	1517	430	0
0	424	1494	2915	4419	5791	6885	7638	8062	8233	8258	8242	8257	8322	8396	8391	8200	7723	6905	5752	4347	2841	1444	407	0
0	484	1685	3239	4837	6252	7354	8112	8565	8797	8898	8944	8981	9016	9021	8938	8687	8185	7367	6212	4768	3169	1638	468	0
0	561	1928	3654	5374	6841	7944	8688	9146	9413	9568	9657	9701	9700	9640	9496	9217	8730	7951	6816	5331	3612	1900	551	0
0	591	2026	3824	5597	7092	8206	8954	9420	9702	9872	9968	10000	9968	9872	9702	9420	8954	8206	7092	5597	3824	2026	591	0
0	551	1900	3612	5331	6816	7951	8730	9217	9496	9640	9700	9701	9657	9568	9413	9146	8688	7944	6841	5374	3654	1928	561	0
0	468	1638	3169	4768	6212	7367	8185	8687	8938	9021	9016	8981	8944	8898	8797	8565	8112	7354	6252	4837	3239	1685	484	0
0	407	1444	2841	4347	5752	6905	7723	8200	8391	8396	8322	8257	8242	8258	8233	8062	7638	6885	5791	4419	2915	1494	424	0
0	430	1517	2962	4493	5888	6992	7731	8113	8211	8139	8018	7943	7953	8026	8080	7996	7654	6966	5908	4541	3014	1553	442	0
0	557	1916	3631	5327	6741	7738	8306	8519	8498	8371	8241	8178	8208	8312	8428	8451	8253	7707	6731	5333	3643	1926	561	0
0	746	2505	4610	6534	7961	8796	9126	9129	8988	8835	8734	8702	8740	8843	8992	9122	9106	8763	7919	6492	4576	2485	739	0
0	903	2989	5393	7460	8834	9473	9559	9368	9138	8990	8938	8943	8976	9053	9204	9415	9568	9440	8769	7381	5324	2945	889	0
0	932	3056	5447	7418	8622	9055	8960	8655	8391	8272	8269	8301	8324	8364	8491	8730	8987	9028	8549	7325	5363	3002	914	0
0	780	2542	4498	6066	6965	7217	7047	6741	6511	6432	6457	6498	6509	6520	6607	6815	7077	7197	6903	5984	4423	2494	764	0
0	479	1558	2743	3676	4189	4301	4163	3957	3813	3773	3800	3830	3834	3831	3876	4006	4185	4290	4151	3625	2696	1527	469	0
0	156	505	887	1184	1342	1370	1319	1248	1201	1190	1201	1212	1212	1209	1222	1264	1326	1367	1330	1168	872	495	152	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

THE EXECUTION TIME FOR THIS PROBLEM WAS C MINUTES, 30 SECONDS.

```
6     SAMPLE PROBLEM - SHEAR BUCKLING                                         004602P010001
6 ++2+ +3+2 +5+10 +1+1 ++2+8 ++1 +1+1+1 ++++++++++++++                        004602P010002
6 +9 +16.45 +12                                                               004602P010003
6 +21000000 +1700000 +650000 +.21 +.007                                       004602P010004
6 +45-45+45-45-45+45-45+45                                                    004602P010005
6 +++1                                                                        004602P010006
```

```
     SAMPLE PROBLEM - SHEAR BUCKLING

THE BOUNDARY CONDITIONS AT X=0 AND X=A ARE
CLAMPED, CLAMPED

THE BOUNDARY CONDITIONS AT Y=0 AND Y=B ARE
SIMPLE, SIMPLE

THERE ARE  5 MODES IN THE X DIRECTION, STARTING WITH M =  1 .
THERE ARE 10 MODES IN THE Y DIRECTION, STARTING WITH N =  1 .

THE STIFFNESS MATRIX SIZE IS 150 BY 150

A STABILITY SOLUTION WILL BE SOUGHT

A =              9.00000

B =             16.45000

R =             12.00000

MU =             0.0

FOR THE  8 PLY LAMINATE

E1 =         0.210000E 08

E2 =         0.170000E 07

G =         0.650000E 06

NU12 = 0.2100

H(I) =   0.0070

T =    0.0560

THE ORIENTATIONS ARE
   45.0000

  -45.0000

   45.0000

  -45.0000

  -45.0000

   45.0000

  -45.0000
```

45.0000

```
THE CONSTITUTIVE MATRIX IS

  0.3653700E 06   0.2925696E 06   0.0             0.3662109E-02   0.1220703E-03   0.1220703E-03
  0.2925696E 06   0.3653695E 06   0.0             0.1220703E-03   0.2929688E-02   0.1220703E-03
  0.0             0.0             0.3089061E 06   0.1220703E-03   0.1220703E-03   0.7324219E-03
  0.3662109E-02   0.1220703E-03   0.1220703E-03   0.9548331E 02   0.7645819E 02   0.2657442E 02
  0.1220703E-03   0.2929688E-02   0.1220703E-03   0.7645819E 02   0.9548315E 02   0.2657440E 02
  0.1220703E-03   0.1220703E-03   0.7324219E-03   0.2657442E 02   0.2657440E 02   0.8072745E 02

THE LAMINATE PROPERTIES ARE

EX =    0.234098E 07   EY =    0.234098E 07   G =    0.551618E 07   NUXY =   0.8008   NUYX =   0.8007

PX(I,J) FOLLOWS
 0.0

PY(I,J) FOLLOWS
 0.0

PXY(I,J) FOLLOWS
 1.0000E 00

THE BUCKLING EIGENVALUE IS    0.5006819E 03 FOR M =  1, N =  6.

THE CONTRIBUTIONS OF THE SERIES TERMS FOR W FOLLOW
 -0.2243E-11  0.7644E-02 -0.1333E-10  0.4221E-01 -0.2644E-09  0.1142E 01  0.6323E-09 -0.6329E 00 -0.1261E-09 -0.1098E-01
 -0.8917E-02 -0.5158E-11 -0.3619E-01 -0.4761E-11 -0.2724E 00 -0.4363E-09  0.1000E 01  0.4095E-09 -0.2227E 00 -0.1120E-10
  0.3574E-11 -0.1415E-01  0.7597E-11 -0.1920E-01  0.2547E-10 -0.2264E 00 -0.2083E-09  0.3338E 00  0.9032E-10 -0.1029E-01
  0.1386E-02  0.3442E-11 -0.3647E-02  0.9592E-11 -0.4007E-01 -0.4753E-10  0.1214E-01  0.2539E-11  0.4119E-01  0.8886E-11
  0.1435E-11 -0.5456E-03  0.5684E-11  0.3906E-02  0.3032E-10 -0.4466E-01 -0.8372E-10  0.4226E-01  0.1792E-10  0.8568E-02
```

GENERAL DYNAMICS CONVAIR AEROSPACE DIVISION FORT WORTH OPERATION
370 PROCEDURE SS8 PROBLEM 004602-01 05/04/73 PAGE 0004

THE W DEFLECTIONS DIVIDED BY 0.471940E 01/10000 FOLLOW

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	238	233	-34	-336	-390	-116	282	480	305	-111	-439	-424	-98	264	388	222	-50	-206	-171	-40	52	62	30	0
0	809	831	-30	-1067	-1347	-523	818	1598	1147	-209	-1396	-1488	-475	772	1295	833	-66	-649	-606	-201	131	207	119	0
0	1508	1641	152	-1807	-2559	-1282	1181	2894	2401	17	-2370	-2884	-1231	1134	2357	1744	134	-1078	-1187	-528	135	374	254	0
0	2162	2520	601	-2255	-3746	-2389	1065	3977	3906	768	-2960	-4322	-2390	1060	3267	2846	667	-1302	-1803	-1037	-12	506	417	0
0	2651	3343	1319	-2241	-4683	-3748	337	4574	5462	2079	-2925	-5549	-3880	414	3807	4008	1555	-1204	-2351	-1696	-343	553	574	0
0	2912	4005	2226	-1737	-5218	-5181	-949	4548	6840	3814	-2200	-6365	-5530	-784	3862	5085	2726	-751	-2742	-2435	-847	474	685	0
0	2935	4435	3185	-839	-5281	-6462	-2593	3894	7820	5719	-871	-6633	-7108	-2388	3403	5929	4044	22	-2906	-3159	-1484	245	709	0
0	2750	4593	4041	285	-4875	-7382	-4323	2711	8236	7498	878	-6290	-8378	-4187	2469	6417	5349	1051	-2793	-3772	-2196	-145	614	0
0	2406	4475	4665	1455	-4070	-7799	-5881	1161	8009	8895	2829	-5347	-9157	-5971	1136	6464	6500	2255	-2380	-4198	-2930	-686	388	0
0	1953	4107	4986	2520	-2977	-7674	-7081	-569	7165	9748	4772	-3886	-9344	-7562	-493	6036	7389	3555	-1671	-4390	-3632	-1350	40	0
0	1440	3542	4994	3380	-1730	-7060	-7827	-2301	5815	10000	6530	-2043	-8925	-8830	-2298	5151	7943	4864	-701	-4323	-4251	-2090	-404	0
0	910	2845	4727	3987	-464	-6076	-8105	-3872	4131	9671	7955	0	-7955	-9671	-4131	3872	8105	6076	464	-3987	-4727	-2845	-910	0
0	404	2090	4251	4323	701	-4864	-7943	-5151	2298	8830	8925	2043	-6530	*****	-5815	2301	7827	7060	1730	-3380	-4994	-3542	-1440	0
0	-40	1350	3632	4390	1671	-3555	-7389	-6036	493	7562	9344	3886	-4772	-9748	-7165	569	7081	7674	2977	-2520	-4986	-4107	-1953	0
0	-388	686	2930	4198	2380	-2255	-6500	-6464	-1136	5971	9157	5347	-2829	-8895	-8009	-1161	5881	7799	4070	-1455	-4665	-4475	-2406	0
0	-614	145	2196	3772	2793	-1051	-5349	-6417	-2469	4187	8378	6290	-878	-7498	-8236	-2711	4323	7382	4875	-285	-4041	-4593	-2750	0
0	-709	-245	1484	3159	2906	-22	-4044	-5929	-3403	2388	7108	6633	871	-5719	-7820	-3894	2593	6462	5281	839	-3185	-4435	-2935	0
0	-685	-474	847	2435	2742	751	-2726	-5085	-3862	784	5530	6365	2200	-3814	-6840	-4548	949	5181	5218	1737	-2226	-4005	-2912	0
0	-574	-553	343	1696	2351	1204	-1555	-4008	-3807	-414	3880	5549	2925	-2079	-5462	-4574	-337	3748	4683	2241	-1319	-3343	-2651	0
0	-417	-506	12	1037	1803	1302	-667	-2846	-3267	-1060	2390	4322	2960	-768	-3906	-3977	-1065	2389	3746	2255	-601	-2520	-2162	0
0	-254	-374	-135	528	1187	1078	-134	-1744	-2357	-1134	1231	2884	2370	-17	-2401	-2894	-1181	1282	2559	1807	-152	-1641	-1508	0
0	-119	-207	-131	201	606	649	66	-833	-1295	-772	475	1488	1396	209	-1147	-1598	-818	523	1347	1067	30	-831	-809	0
0	-30	-62	-52	40	171	206	50	-222	-388	-264	98	424	439	111	-305	-480	-282	116	390	336	34	-233	-238	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

THE BUCKLING EIGENVALUE IS -0.7427390E 03 FOR M = 1, N = 6.

THE CONTRIBUTIONS OF THE SERIES TERMS FOR W FOLLOW

0.3631E-11	-0.2483E-02	0.2058E-10	-0.6968E-01	0.5375E-09	0.1000E 01	-0.4853E-09	-0.2751E 00	0.1554E-11	-0.3611E-01
0.9910E-02	0.9287E-11	0.3884E-01	0.6842E-10	0.3470E 00	-0.4849E-09	-0.6145E 00	0.1627E-09	0.3848E-02	0.2147E-10
-0.5746E-11	0.1553E-02	-0.1276E-10	0.2179E-02	-0.4942E-10	-0.1588E 00	0.1306E-09	0.1111E 00	-0.3316E-11	0.2021E-01
-0.1692E-02	0.1568E-11	-0.6809E-03	0.1755E-10	0.4223E-01	-0.4988E-10	-0.3605E-01	0.4644E-11	-0.6943E-02	-0.1012E-13
-0.1235E-11	0.1412E-02	-0.3039E-11	0.4844E-02	-0.2492E-10	-0.2717E-01	0.4598E-10	0.1869E-01	0.5196E-12	0.4790E-02

THE W DEFLECTIONS DIVIDED BY 0.327346E 01/10000 FOLLOW

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	-43	-30	63	190	237	112	-149	-370	-363	-94	266	470	373	33	-328	-475	-310	47	335	346	102	-155	-191	0
0	-140	-70	272	688	786	294	-605	-1293	-1169	-194	1011	1613	1176	-26	-1210	-1600	-946	280	1191	1145	280	-581	-673	0
0	-241	-49	645	1381	1411	333	-1367	-2489	-2021	-55	2142	3039	1969	-387	-2482	-2946	-1503	810	2347	2062	358	-1216	-1313	0
0	-297	90	1183	2156	1913	77	-2395	-3707	-2597	489	3540	4408	2405	-1177	-3969	-4148	-1673	1689	3593	2817	206	-1992	-1994	0
0	-267	386	1856	2900	2150	-540	-3606	-4736	-2696	1487	5052	5457	2281	-2406	-5481	-4947	-1294	2881	4743	3226	-231	-2835	-2622	0
0	-120	848	2610	3511	2050	-1493	-4870	-5427	-2250	2867	6501	6018	1553	-3962	-6829	-5201	-366	4265	5648	3208	-936	-3664	-3130	0
0	157	1459	3373	3908	1604	-2685	-6045	-5696	-1308	4467	7717	6026	312	-5651	-7847	-4893	978	5661	6206	2781	-1821	-4401	-3481	0
0	561	2179	4071	4041	852	-3984	-7001	-5524	4	6093	8563	5500	-1264	-7253	-8420	-4094	2537	6883	6369	2029	-2767	-4976	-3657	0
0	1066	2954	4640	3893	-132	-5263	-7646	-4942	1537	7566	8958	4521	-2982	-8578	-8498	-2927	4098	7783	6138	1070	-3653	-5337	-3657	0
0	1629	3719	5035	3478	-1266	-6414	-7935	-4014	3143	8755	8879	3199	-4664	-9506	-8094	-1534	5494	8275	5556	17	-4385	-5453	-3490	0
0	2201	4408	5223	2833	-2461	-7354	-7854	-2819	4700	9581	8353	1654	-6178	-9981	-7268	-41	6615	8344	4693	-1030	-4903	-5321	-3172	0
0	2731	4960	5183	2000	-3632	-8018	-7410	-1439	6106	10000	7432	0	-7432	*****	-6106	1439	7410	8018	3632	-2000	-5183	-4960	-2731	0
0	3172	5321	4903	1030	-4693	-8344	-6616	41	7268	9981	6178	-1654	-8353	-9581	-4700	2819	7854	7354	2461	-2833	-5223	-4408	-2201	0
0	3490	5453	4385	-17	-5556	-8275	-5494	1534	8094	9506	4664	-3199	-8879	-8755	-3143	4014	7935	6414	1266	-3478	-5035	-3719	-1629	0
0	3657	5337	3653	-1070	-6138	-7783	-4098	2927	8498	8578	2982	-4521	-8958	-7566	-1537	4942	7646	5263	132	-3893	-4640	-2954	-1066	0
0	3657	4976	2767	-2029	-6369	-6883	-2537	4094	8420	7253	1264	-5500	-8563	-6093	-4	5524	7001	3984	-852	-4041	-4071	-2179	-561	0
0	3481	4401	1821	-2781	-6206	-5661	-978	4893	7847	5651	-312	-6026	-7717	-4467	1308	5696	6045	2685	-1604	-3908	-3373	-1459	-157	0
0	3130	3664	936	-3208	-5648	-4265	366	5201	6829	3962	-1553	-6018	-6501	-2867	2250	5427	4870	1493	-2050	-3511	-2610	-848	120	0
0	2622	2835	231	-3226	-4743	-2881	1294	4947	5481	2406	-2281	-5457	-5052	-1487	2696	4736	3606	540	-2150	-2900	-1856	-386	267	0
0	1994	1992	-206	-2817	-3593	-1689	1673	4148	3969	1177	-2405	-4408	-3540	-489	2597	3707	2395	-77	-1913	-2156	-1183	-90	297	0
0	1313	1216	-358	-2062	-2347	-810	1503	2946	2482	387	-1969	-3039	-2142	55	2021	2489	1367	-333	-1411	-1381	-645	49	241	0
0	673	581	-280	-1145	-1191	-280	946	1600	1210	26	-1176	-1613	-1011	194	1169	1293	605	-294	-786	-688	-272	70	140	0
0	191	155	-102	-346	-335	-47	310	475	328	-33	-373	-470	-266	94	363	370	149	-112	-237	-190	-63	30	43	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

THE EXECUTION TIME FOR THIS PROBLEM WAS 2 MINUTES, 48 SECONDS.

```
6     SAMPLE PROBLEM - PANEL VIBRATION                                004602P020001
6 +1++ +4+2 +3+3 +1+1 +3 +2+2++1 +++++++++++++++++                    004602P020002
6 +6 +4 +20                                                           004602P020003
6 +30000000 +2700000 +650000 +.21 +.0053  +45-45                      004602P020004
6 +.00018                                                             004602P020005
```

```
     SAMPLE PROBLEM - PANEL VIBRATION

THE BOUNDARY CONDITIONS AT X=0 AND X=A ARE
CLAMPED, FREE

THE BOUNDARY CONDITIONS AT Y=0 AND Y=B ARE
SIMPLE, SIMPLE

THERE ARE  3 MODES IN THE X DIRECTION, STARTING WITH M =  1 .
THERE ARE  3 MODES IN THE Y DIRECTION, STARTING WITH N =  1 .

THE STIFFNESS MATRIX SIZE IS  27 BY  27

A DYNAMIC SOLUTION WILL BE SOUGHT

A =               6.00000

B =               4.00000

R =              20.00000

MU =              0.0

FOR THE  2 PLY LAMINATE

E1 =         0.300000E 08

E2 =         0.270000E 07

G =         0.650000E 06

NU12 = 0.2100

H(I) =   0.0053

T =    0.0106

THE ORIENTATIONS ARE
   45.0000

  -45.0000
```

```
GENERAL DYNAMICS          CONVAIR AEROSPACE DIVISION          FORT WORTH OPERATION
370 PROCEDURE SS8              PROBLEM 004602-02              05/04/73   PAGE 0002

THE CONSTITUTIVE MATRIX IS

  0.9690731E 05    0.8312731E 05    0.0              0.0              0.0              -0.1924780E 03
  0.8312731E 05    0.9690725E 05    0.0              0.0              0.0              -0.1924779E 03
  0.0              0.0              0.8398313E 05   -0.1924780E 03   -0.1924779E 03     0.0
  0.0              0.0             -0.1924780E 03    0.9073753E 00    0.7783483E 00     0.0
  0.0              0.0             -0.1924779E 03    0.7783483E 00    0.9073747E 00     0.0
 -0.1924780E 03   -0.1924779E 03    0.0              0.0              0.0               0.7863617E 00

THE LAMINATE PROPERTIES ARE

EX =   0.241514E 07   EY =   0.241514E 07   G =   0.792294E 07   NUXY =  0.8578   NUYX =  0.8578

THE MATERIAL DENSITY =  0.18000000E-03 LB.-SEC.**2/IN.**4
```

FREQUENCY	M	N
0.17837E 03	1	1
0.22114E 03	1	2
0.44064E 03	2	2
0.53527E 03	1	3
0.69488E 03	2	3
0.96692E 03	2	1
0.99399E 03	2	1
0.10902E 04	3	3
0.14863E 04	3	1
0.13607E 05	2	1
0.18098E 05	1	1
0.25587E 05	3	2
0.30574E 05	1	1
0.37216E 05	2	2
0.39052E 05	1	1
0.48045E 05	3	3
0.51674E 05	1	2
0.57091E 05	2	1
0.62332E 05	2	3
0.63678E 05	1	2
0.71649E 05	3	1
0.77191E 05	1	3
0.80854E 05	2	2
0.90266E 05	1	3
0.96942E 05	3	2

0.10690E 06	2	3
0.12326E 06	3	3

THE FREQUENCY IS 0.1783669E 03 CPS. FOR M = 1, N = 1.

THE CONTRIBUTIONS OF THE SERIES TERMS FOLLOW

-0.1106E-01	-0.8295E-03	-0.4571E-04	0.9215E-03	-0.1925E-04	0.5167E-05	0.1470E-03	0.2947E-05	-0.2778E-05	0.1866E-01
0.1068E-02	-0.5834E-04	-0.1127E-02	0.1236E-03	0.1372E-04	-0.1871E-03	-0.6356E-06	0.3680E-05	0.9776E 00	0.1939E 00
-0.2981E-01	0.4485E-01	0.2445E-01	0.3505E-02	0.5183E-01	0.1346E-02	-0.1478E-03			

THE W DEFLECTIONS DIVIDED BY 0.299838E 01/10000 FOLLOW

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	10	20	29	37	45	51	56	59	61	61	60	58	55	51	46	41	36	30	25	20	14	9	4	0
0	38	75	109	141	169	192	210	222	229	231	227	219	207	192	174	155	135	114	94	74	54	36	17	0
0	79	157	230	297	355	403	441	467	481	484	476	459	433	401	363	322	280	237	194	152	112	74	36	0
0	132	260	381	490	587	666	728	771	794	799	785	756	712	658	595	527	457	385	315	247	182	119	59	0
0	191	376	551	710	849	964	1054	1115	1148	1153	1133	1089	1025	945	854	754	651	548	447	349	257	168	83	0
0	254	500	733	944	1129	1282	1400	1481	1525	1531	1502	1442	1355	1247	1124	991	853	716	582	453	332	217	107	0
0	318	628	919	1184	1416	1608	1756	1857	1910	1916	1878	1801	1690	1552	1395	1226	1053	880	713	554	405	264	130	0
0	383	755	1105	1424	1702	1933	2110	2231	2294	2300	2252	2157	2020	1852	1660	1455	1245	1037	837	648	472	307	151	0
0	446	879	1287	1658	1983	2252	2458	2599	2671	2676	2618	2504	2342	2143	1916	1675	1428	1185	953	735	533	346	169	0
0	507	1000	1464	1887	2256	2563	2798	2958	3039	3044	2976	2843	2656	2425	2164	1886	1603	1325	1061	815	589	381	186	0
0	566	1117	1636	2110	2524	2867	3131	3310	3401	3406	3329	3178	2965	2703	2407	2092	1773	1461	1165	892	642	414	202	0
0	625	1233	1806	2329	2788	3169	3462	3661	3763	3769	3682	3514	3276	2983	2652	2300	1944	1597	1270	968	695	447	218	0
0	683	1348	1976	2550	3053	3473	3797	4018	4132	4139	4046	3860	3597	3273	2907	2518	2124	1741	1381	1050	752	482	235	0
0	743	1466	2150	2775	3326	3786	4143	4387	4515	4527	4427	4226	3939	3584	3182	2754	2321	1900	1504	1142	816	523	254	0
0	804	1588	2330	3010	3610	4114	4506	4777	4922	4939	4835	4619	4308	3922	3483	3015	2540	2078	1645	1248	891	570	277	0
0	869	1716	2520	3258	3911	4461	4893	5194	5358	5383	5276	5046	4712	4294	3817	3306	2788	2282	1807	1371	979	627	305	0
0	938	1852	2721	3521	4232	4832	5306	5640	5826	5863	5754	5512	5155	4704	4187	3633	3067	2514	1993	1514	1082	693	338	0
0	1010	1997	2935	3801	4573	5228	5748	6118	6330	6379	6272	6018	5637	5153	4595	3994	3378	2774	2203	1677	1201	770	375	0
0	1087	2149	3161	4097	4933	5646	6216	6626	6866	6931	6825	6560	6156	5638	5038	4388	3719	3062	2437	1859	1333	857	418	0
0	1167	2309	3398	4407	5312	6086	6708	7161	7431	7513	7410	7135	6708	6156	5511	4810	4087	3372	2690	2057	1478	952	465	0
0	1251	2474	3643	4728	5703	6542	7219	7716	8017	8118	8020	7735	7284	6697	6008	5254	4474	3700	2958	2267	1633	1053	515	0
0	1336	2643	3893	5057	6104	7008	7742	8284	8619	8739	8645	8351	7877	7254	6520	5713	4874	4039	3237	2486	1794	1159	567	0
0	1422	2814	4147	5389	6510	7481	8272	8860	9228	9368	9280	8976	8478	7820	7040	6180	5282	4385	3521	2709	1958	1266	620	0
0	1508	2986	4401	5723	6918	7955	8803	9438	9840	10000	9916	9603	9083	8389	7562	6648	5692	4733	3806	2933	2123	1375	674	0

```
THE FREQUENCY IS    0.2211411E 03 CPS. FOR M =  1, N =  2.

THE CONTRIBUTIONS OF THE SERIES TERMS FOLLOW
  0.4073E-02 -0.3105E-02 -0.4223E-03 -0.3826E-03 -0.9828E-05  0.6660E-04 -0.3544E-04  0.9751E-04 -0.2021E-04 -0.3748E-02
  0.4950E-02  0.7037E-04 -0.1562E-03  0.1002E-03  0.2738E-04  0.1240E-03 -0.1352E-03  0.1602E-04 -0.1920E 00  0.9796E 00
  0.8096E-02 -0.3797E-01  0.4220E-01  0.3574E-02 -0.1021E-01  0.1279E-02  0.8947E-03
```

THE W DEFLECTIONS DIVIDED BY 0.297272E 01/10000 FOLLOW

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	7	14	20	23	25	24	21	16	9	0	-8	-17	-26	-34	-40	-45	-47	-46	-43	-37	-30	-21	-10	0
0	29	56	78	93	100	97	86	66	38	6	-29	-66	-101	-131	-156	-173	-180	-178	-166	-145	-117	-81	-42	0
0	64	123	172	205	220	216	191	148	90	19	-58	-138	-214	-281	-335	-371	-388	-385	-360	-315	-253	-176	-91	0
0	112	214	298	357	384	378	337	264	164	43	-89	-226	-357	-473	-566	-630	-660	-655	-613	-537	-432	-302	-155	0
0	170	326	455	545	588	580	520	412	263	81	-118	-324	-522	-697	-839	-937	-985	-978	-917	-805	-647	-452	-233	0
0	238	458	638	766	829	820	740	592	386	136	-140	-426	-701	-946	-1145	-1283	-1351	-1345	-1263	-1109	-892	-625	-321	0
0	316	606	847	1017	1103	1095	993	802	535	209	-153	-527	-889	-1211	-1474	-1658	-1752	-1746	-1642	-1444	-1162	-814	-419	0
0	401	770	1076	1294	1406	1401	1277	1041	708	301	-153	-624	-1079	-1487	-1820	-2055	-2177	-2174	-2047	-1802	-1452	-1018	-524	0
0	493	946	1324	1595	1736	1735	1589	1306	904	411	-140	-713	-1268	-1768	-2177	-2468	-2620	-2623	-2473	-2179	-1757	-1232	-635	0
0	590	1134	1587	1915	2089	2093	1925	1594	1122	539	-114	-794	-1454	-2050	-2540	-2890	-3076	-3085	-2913	-2570	-2074	-1455	-750	0
0	692	1330	1864	2251	2459	2471	2282	1902	1357	682	-75	-866	-1636	-2331	-2905	-3318	-3541	-3558	-3364	-2970	-2399	-1684	-868	0
0	797	1533	2150	2599	2844	2864	2654	2226	1608	839	-26	-930	-1812	-2611	-3272	-3751	-4012	-4037	-3822	-3378	-2730	-1917	-988	0
0	905	1742	2444	2957	3240	3270	3040	2563	1870	1007	32	-988	-1984	-2889	-3640	-4185	-4487	-4522	-4286	-3791	-3066	-2154	-1111	0
0	1015	1954	2743	3321	3644	3684	3434	2908	2140	1181	96	-1040	-2153	-3165	-4007	-4621	-4964	-5010	-4753	-4207	-3404	-2393	-1234	0
0	1126	2168	3045	3690	4053	4103	3833	3258	2415	1360	164	-1090	-2321	-3441	-4375	-5059	-5443	-5501	-5224	-4627	-3746	-2633	-1358	0
0	1238	2383	3349	4061	4464	4525	4235	3611	2693	1541	233	-1140	-2488	-3718	-4745	-5498	-5925	-5994	-5696	-5048	-4088	-2875	-1484	0
0	1350	2599	3653	4432	4875	4947	4638	3964	2970	1721	302	-1191	-2557	-3996	-5116	-5940	-6409	-6489	-6171	-5472	-4433	-3118	-1609	0
0	1462	2815	3957	4803	5286	5368	5039	4316	3245	1899	368	-1244	-2829	-4277	-5490	-6384	-6895	-6986	-6647	-5896	-4778	-3362	-1735	0
0	1573	3030	4261	5172	5696	5788	5438	4665	3518	2075	431	-1300	-3003	-4561	-5867	-6830	-7383	-7485	-7125	-6322	-5125	-3606	-1861	0
0	1684	3244	4563	5541	6103	6206	5835	5011	3789	2247	492	-1359	-3181	-4848	-6246	-7279	-7874	-7986	-7605	-6750	-5472	-3851	-1988	0
0	1795	3458	4864	5908	6509	6621	6229	5355	4056	2417	549	-1421	-3361	-5137	-6628	-7730	-8366	-8489	-8085	-7178	-5820	-4096	-2115	0
0	1906	3671	5164	6274	6914	7035	6622	5698	4322	2585	605	-1484	-3542	-5428	-7011	-8182	-8859	-8992	-8567	-7606	-6168	-4342	-2241	0
0	2016	3884	5464	6639	7319	7449	7014	6040	4587	2752	660	-1549	-3725	-5719	-7394	-8635	-9353	-9496	-9048	-8035	-6517	-4588	-2368	0
0	2127	4097	5764	7004	7723	7862	7406	6381	4851	2919	714	-1614	-3908	-6011	-7779	-9088	-9847	*****	-9530	-8464	-6865	-4833	-2495	0

```
THE FREQUENCY IS    0.4406414E 03 CPS. FOR M =  2, N =  2.

THE CONTRIBUTIONS OF THE SERIES TERMS FOLLOW
  0.2342E-02  0.6560E-03 -0.4196E-03 -0.1309E-04 -0.2959E-02 -0.2190E-03 -0.7851E-04  0.3729E-03  0.4559E-04  0.5082E-03
  0.4210E-03 -0.2247E-03  0.9733E-03  0.5467E-02  0.1417E-03 -0.6676E-03 -0.2815E-03  0.1048E-03 -0.1788E-01 -0.4466E-01
 -0.3510E-01  0.5143E-01  0.9952E 00  0.2964E-01 -0.3236E-01  0.3157E-01  0.2215E-01
```

GENERAL DYNAMICS CONVAIR AEROSPACE DIVISION FORT WORTH OPERATION
370 PROCEDURE SS8 PROBLEM 004602-02 05/04/73 PAGE 0010

THE W DEFLECTIONS DIVIDED BY 0.314027E 01/10000 FOLLOW

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	48	94	131	159	175	177	166	144	111	70	26	-19	-63	-101	-132	-154	-166	-167	-157	-139	-111	-78	-40	0
0	180	347	487	589	647	657	618	534	414	265	101	-67	-228	-371	-487	-569	-613	-618	-584	-514	-414	-290	-149	0
0	372	716	1005	1217	1338	1359	1281	1111	863	559	219	-128	-462	-759	-1000	-1172	-1265	-1276	-1208	-1065	-859	-602	-309	0
0	602	1158	1626	1972	2169	2207	2083	1812	1414	923	375	-188	-730	-1214	-1608	-1890	-2046	-2068	-1960	-1730	-1396	-979	-504	0
0	849	1633	2295	2784	3067	3124	2956	2578	2022	1333	561	-235	-1003	-1690	-2253	-2658	-2884	-2922	-2773	-2452	-1980	-1389	-716	0
0	1092	2103	2957	3591	3961	4042	3833	3354	2645	1761	769	-258	-1252	-2145	-2879	-3412	-3714	-3771	-3586	-3175	-2567	-1802	-929	0
0	1316	2535	3566	4335	4787	4894	4653	4086	3241	2182	989	-252	-1456	-2542	-3440	-4097	-4474	-4555	-4340	-3849	-3116	-2190	-1129	0
0	1503	2897	4079	4964	5491	5625	5362	4728	3773	2570	1209	-213	-1598	-2853	-3897	-4665	-5115	-5222	-4986	-4430	-3591	-2526	-1303	0
0	1642	3166	4462	5437	6024	6186	5915	5237	4206	2900	1415	-143	-1668	-3056	-4218	-5081	-5593	-5728	-5484	-4881	-3962	-2790	-1440	0
0	1724	3325	4689	5722	6352	6538	6272	5578	4511	3150	1594	-47	-1661	-3139	-4384	-5316	-5880	-6042	-5800	-5173	-4206	-2965	-1532	0
0	1740	3359	4743	5796	6448	6654	6406	5726	4664	3300	1730	64	-1581	-3098	-4383	-5355	-5953	-6141	-5912	-5285	-4304	-3038	-1571	0
0	1690	3264	4614	5648	6297	6518	6300	5660	4646	3332	1809	183	-1434	-2933	-4214	-5192	-5804	-6013	-5807	-5204	-4246	-3001	-1553	0
0	1571	3038	4300	5274	5895	6123	5943	5371	4446	3234	1818	295	-1230	-2654	-3882	-4829	-5433	-5655	-5481	-4926	-4028	-2851	-1476	0
0	1387	2685	3807	4680	5247	5472	5339	4857	4059	2999	1747	389	-982	-2274	-3398	-4277	-4849	-5075	-4940	-4454	-3651	-2588	-1342	0
0	1142	2213	3145	3879	4367	4578	4496	4125	3487	2623	1590	455	-704	-1808	-2781	-3552	-4066	-4286	-4195	-3797	-3122	-2218	-1151	0
0	842	1636	2333	2891	3276	3461	3432	3188	2739	2111	1343	483	-409	-1274	-2049	-2676	-3108	-3310	-3264	-2973	-2455	-1749	-909	0
0	495	965	1388	1739	1998	2147	2172	2066	1829	1470	1008	470	-108	-687	-1223	-1672	-1998	-2170	-2171	-1999	-1664	-1192	-622	0
0	108	219	333	450	564	665	743	784	778	715	592	413	189	-64	-324	-566	-763	-894	-942	-900	-769	-560	-295	0
0	-309	-588	-807	-947	-995	-950	-822	-627	-390	-137	105	316	478	582	627	619	568	489	395	299	210	131	62	0
0	-751	-1441	-2014	-2426	-2648	-2668	-2491	-2140	-1650	-1067	-439	183	757	1243	1617	1862	1972	1953	1815	1576	1253	869	445	0
0	-1207	-2324	-3265	-3962	-4367	-4457	-4234	-3724	-2976	-2052	-1026	24	1025	1912	2630	3142	3424	3472	3291	2905	2341	1640	844	0
0	-1674	-3225	-4542	-5531	-6124	-6288	-6021	-5352	-4342	-3072	-1641	-153	1286	2583	3656	4444	4905	5023	4802	4265	3456	2430	1253	0
0	-2144	-4136	-5832	-7116	-7901	-8141	-7829	-7001	-5728	-4110	-2270	-341	1542	3255	4688	5756	6400	6590	6328	5642	4584	3230	1668	0
0	-2616	-5048	-7126	-8706	-9683	*****	-9645	-8657	-7120	-5153	-2903	-532	1797	3927	5722	7071	7898	8162	7860	7023	5716	4032	2084	0

THE EXECUTION TIME FOR THIS PROBLEM WAS 0 MINUTES, 15 SECONDS.

A P P E N D I X I V

P R O G R A M L I S T I N G S

```
C      CONTROL PROGRAM FOR THE ANALYSIS OF ANISOTROPIC CURVED PANELS.     SS8A000
C                                                                         SS8A001
       CALL GSTART ('SS8',IDIOT)                                          SS8A002
     1 CALL READ                                                          SS8A003
       CALL TABLE                                                         SS8A004
       CALL ASEMBL                                                        SS8A005
       CALL SOLVE                                                         SS8A006
       CALL PRINT                                                         SS8A007
       GO TO 1                                                            SS8A008
       END                                                                SS8A009

                                                          CC = 00010
```

```
      SUBROUTINE READ
C
C **  THIS SUBROUTINE READS ALL THE NECESSARY INPUT DATA, MAKES DATA
C **  CHECKS, AND WRITES PRELIMINARY DATA.
C
      DIMENSION     YSTRNG(100),   YBARS(100),    ZBARS(100),    AS(100),
     1              XIYYS(100),    XIYZS(100),    XIZZS(100),    ES(100),
     2              GJS(100),      RHOS(100),     PAXS(100),
     3              XRINGS(50),    XBARR(50),     ZBARR(50),     AR(50),
     4              XIXXR(50),     XIXZR(50),     XIZZR(50),     ER(50),
     5              GJR(50),       RHOR(50),      PAXR(50),
     8              PMASS(50),     IPWW(50),      IPWY(50),
     9              PX(10,10),     PY(10,10),     PXY(10,10),
     C              PC(50),        IPXX(50),      IPYY(50),
     D              FC(50),        IFXX(50),      IFYY(50),
     E              ITAGCM(50),    Q (10,10),
     F              PLMOM(50),     ITAGLM(50),    IDISLM(50),
     G              PKC(50),       IGSPRX(50),    IGSPRY(50),
     H              PLINE(50),     IDISLS(50),    ITAGLS(50)
      DIMENSION     ITIME(12),     TIME(50)
      DIMENSION     AMAT(3,3),     BMAT(3,3),     DMAT(3,3),     H(40),
     1              THETA(40),     E1(40),        E2(40),        G(40),
     2              XNU12(40)
      DIMENSION     EC(3,40),      ET(3,40),      ANGCK(3,10),   MCHK(3)
      DIMENSION     V(2,10),       PRTNX(5,5),    PRTNY(5,5),
     1              PRTNXY(5,5),   PRTQ(5,5)
      DIMENSION     AI(3,3),       A(3,3)
      COMMON                 U(50,50)
      COMMON / CHECKS / IERROR
      COMMON / CNTROL / IFLAGD,    IFLAGB,    IFLAGW,    IBCX,      IBCY,
     1              IMATL,     IEDGE,     IREACT,    IOUT,      IPRTN,
     2              IPRTQ,     IELAST,    INTPRT,    IKDF,      IFLEX
      COMMON / NUMBER / NPLYS,     NTUX,      NTVX,      NTWX,      NTUY,
     1                  NTVY,      NTWY,      NMODES,    NSTRNG,    NRING,
     2                  NPNX,      NPNY,      NQTX,NQTY,NPTLDS,     NPTMOM,
     3                  NLNMOM,    NLMASS,    NPTSUP,               NLNSPR,
     4                  MATSIZ,    MUVSIZ,    MWSIZ,     ITX,       ITY
      COMMON / GEOM   / AA,        BB,        RR,        ALFAX,     ALFAY,
     1              BETAX,     BETAY,     MU
      COMMON / $TIME  / TIME,      ITIME
      COMMON / ABD    / AMAT,      BMAT,      DMAT,      RHAB,      THETA,
     1                  H,         E1,        E2,        G,         XNU12,
     2                  EC,        ET,        ANGCK,     MCHK
      COMMON / PARAM  / YBARS,     ZBARS,     AS,        XIYYS,     XIYZS,
     1                  XIZZS,     ES,        GJS,       RHOS,      PAXS,
     3                  XBARR,     ZBARR,     AR,        XIXXR,     XIXZR,
     4                  XIZZR,     ER,        GJR,       RHOR,      PAXR,
     6                  PMASS,     IPWW,      IPWY,      PX,        PY,
     7                  PXY,       PC,        IPXX,      IPYY,      FC,
     8                  IFXX,      IFYY,      ITAGCM,    Q ,        PLMOM,
     9                  ITAGLM,    IDISLM,    PKC,       IGSPRX,    IGSPRY,
     A                  PLINE,     IDISLS,    ITAGLS
      COMMON / STFVAL / ESV(10,100),   ESW(10,100),   ESDW(10,100),
     1                  ERU(10,50),    ERW(10,50),    ERDW(10,50),
     2                  YSTRNG,        XRINGS
      COMMON / FLEXBL /            XP(50),    YP(50)
```

```
      EQUIVALENCE  ( U(1),PRTNX(1) ),   ( U(26),PRTNY(1) ),   ( U(51),
     1              PRTNXY(1) ),   ( U(76),PRTQ(1) ),   ( U(101),V(1) )
C
      DATA  XDIR / 'X' /, YDIR / 'Y' /
      DATA KIN / 'INN' /, KOUT / 'OUT' /
      REAL MU
C
    1 CALL PROB
      CALL STATUS (ITIME)
      TIME(1) = .01*ITIME(8)
C **  READ AND WRITE TITLE
      READ (5,2)
    2 FORMAT (1X,65H
     1                 )
      WRITE (6,2)
      CALL FREEFD
    5 FORMAT (1X)
      READ (5,5) XFLAGD, XFLAGB, XFLAGW, XBCX   , XBCY   , XTUX   , XTUY   ,
     1           XTX    , XTY    , XMODES, XMATL  , XPLYS  ,
     2           XREACT, XOUT   , XEDGE  , XPNX   , XPNY   , XPRTN  ,
     3           XQTX   , XQTY   , XPRTQ  , XSTRNG, XRING  , XLMASS, XPTLDS,
     4           XPTMOM, XLNMOM, XPTSUP, XLNSPR, XNTPRT, XFLEX
C **  CONVERT FROM REAL TO INTEGER
      INTPRT = XNTPRT + .1
      IFLAGD = XFLAGD + .1
      IFLAGB = XFLAGB + .1
      IKDF = 0
      IF ( IFLAGB - 2 ) 4,4,3
    3 IKDF = 1
      IFLAGB = IFLAGB - 2
    4 CONTINUE
      IFLAGW = XFLAGW + .1
      IBCX   = XBCX   + .1
      IBCY   = XBCY   + .1
      IMATL  = XMATL  + .1
      IEDGE  = XEDGE  + .1
      IREACT = XREACT + .1
      IOUT   = XOUT   + .1
      IPRTN  = XPRTN  + .1
      IPRTQ  = XPRTQ  + .1
      NPLYS  = XPLYS  + .1
      NTUX   = XTUX   + .1
      NTVX   = NTUX
      NTWX   = NTUX
      NTUY   = XTUY   + .1
      NTVY   = NTUY
      NTWY   = NTUY
      ITX    = XTX    + .1
      ITY    = XTY    + .1
      NMODES = XMODES + .1
      IEQS = 0
      IEQR = 0
      IF ( XSTRNG .LT. 0. )  IEQS = 1
      XSTRNG = ABS ( XSTRNG )
      IF ( XRING .LT. 0. )  IEQR = 1
      XRING = ABS ( XRING )
```

```
      NSTRNG = XSTRNG + .1                                            SS8B112
      NRING  = XRING  + .1                                            SS8B113
      NPNX   = XPNX   + .1                                            SS8B114
      NPNY   = XPNY   + .1                                            SS8B115
      NQTX   = XQTX   + .1                                            SS8B116
      NQTY   = XQTY   + .1                                            SS8B117
      NPTLDS = XPTLDS + .1                                            SS8B118
      NPTMOM = XPTMOM + .1                                            SS8B119
      NLNMOM = XLNMOM + .1                                            SS8B120
      NLMASS = XLMASS + .1                                            SS8B121
      NPTSUP = XPTSUP + .1                                            SS8B122
      NLNSPR = XLNSPR + .1                                            SS8B123
      IFLEX  = XFLEX  + .1                                            SS8B124
C **  TEST THE VALUES READ IN                                          SS8B125
      IERROR = 0                                                      SS8B126
      IF ( IFLAGD .LT. 0 .OR. IFLAGD .GT.   1 ) CALL CHECK ('IFLAGD') SS8B127
      IF ( IFLAGB .LT. 0 .OR. IFLAGB .GT.   2 ) CALL CHECK ('IFLAGB') SS8B128
      IF ( IFLAGW .LT. 0 .OR. IFLAGW .GT.   2 ) CALL CHECK ('IFLAGW') SS8B129
      IF ( IBCX   .LT. 1 .OR. IBCX   .GT.   7 ) CALL CHECK ('IBCX  ') SS8B130
      IF ( IBCY   .LT. 0 .OR. IBCY   .GT.   8 ) CALL CHECK ('IBCY  ') SS8B131
      IF ( IMATL  .LT. 1 .OR. IMATL  .GT.   4 ) CALL CHECK ('IMATL ') SS8B132
      IF ( IEDGE  .LT. 0 .OR. IEDGE  .GT.   2 ) CALL CHECK ('IEDGE ') SS8B133
      IF ( IOUT   .LT. 1 .OR. IOUT   .GT.   9 ) CALL CHECK ('IOUT  ') SS8B134
      IF ( IPRTN  .LT. 0 .OR. IPRTN  .GT.   1 ) CALL CHECK ('IPRTN ') SS8B135
      IF ( IPRTQ  .LT. 0 .OR. IPRTQ  .GT.   1 ) CALL CHECK ('IPRTQ ') SS8B136
      IF ( NPLYS  .LT. 1 .OR. NPLYS  .GT.  40 ) CALL CHECK ('NPLYS ') SS8B137
      IF ( NTUX   .LT. 1 .OR. NTUX   .GT.  10 ) CALL CHECK ('NTUX  ') SS8B138
      IF ( NTVX   .LT. 1 .OR. NTVX   .GT.  10 ) CALL CHECK ('NTVX  ') SS8B139
      IF ( NTWX   .LT. 1 .OR. NTWX   .GT.  10 ) CALL CHECK ('NTWX  ') SS8B140
      IF ( NTUY   .LT. 1 .OR. NTUY   .GT.  10 ) CALL CHECK ('NTUY  ') SS8B141
      IF ( NTVY   .LT. 1 .OR. NTVY   .GT.  10 ) CALL CHECK ('NTVY  ') SS8B142
      IF ( NTWY   .LT. 1 .OR. NTWY   .GT.  10 ) CALL CHECK ('NTWY  ') SS8B143
      IF ( ITX    .LT. 0 .OR. ITX    .GT.  20 ) CALL CHECK ('ITX   ') SS8B144
      IF ( ITY    .LT. 0 .OR. ITY    .GT.  20 ) CALL CHECK ('ITY   ') SS8B145
      IF ( NSTRNG .LT. 0 .OR. NSTRNG .GT. 100 ) CALL CHECK ('NSTRNG') SS8B146
      IF ( NRING  .LT. 0 .OR. NRING  .GT.  50 ) CALL CHECK ('NRING ') SS8B147
      IF ( NPNX   .LT. 0 .OR. NPNX   .GT.  10 ) CALL CHECK ('NPNX  ') SS8B148
      IF ( NPNY   .LT. 0 .OR. NPNY   .GT.  10 ) CALL CHECK ('NPNY  ') SS8B149
      IF ( NQTX   .LT. 0 .OR. NQTX   .GT.  10 ) CALL CHECK ('NQTX  ') SS8B150
      IF ( NQTY   .LT. 0 .OR. NQTY   .GT.  10 ) CALL CHECK ('NQTY  ') SS8B151
      IF ( NPTLDS .LT. 0 .OR. NPTLDS .GT.  50 ) CALL CHECK ('NPTLDS') SS8B152
      IF ( NPTMOM .LT. 0 .OR. NPTMOM .GT.  50 ) CALL CHECK ('NPTMOM') SS8B153
      IF ( NLNMOM .LT. 0 .OR. NLNMOM .GT.  50 ) CALL CHECK ('NLNMOM') SS8B154
      IF ( NLMASS .LT. 0 .OR. NLMASS .GT.  50 ) CALL CHECK ('NLMASS') SS8B155
      IF ( NPTSUP .LT. 0 .OR. NPTSUP .GT.  50 ) CALL CHECK ('NPTSUP') SS8B156
      IF ( NLNSPR .LT. 0 .OR. NLNSPR .GT.  50 ) CALL CHECK ('NLNSPR') SS8B157
      IF ( IFLEX  .LT. 0 .OR. IFLEX  .GT.  50 ) CALL CHECK ('IFLEX ') SS8B158
      MATSIZ = NTUX*NTUY + NTVX*NTVY + NTWX*NTWY                      SS8B159
      IF ( MATSIZ .LT. 1 .OR. MATSIZ .GT. 150 ) CALL CHECK ('MATSIZ') SS8B160
      IF ( NMODES .LT. 0 .OR. NMODES .GT.MATSIZ)CALL CHECK ('NMODES') SS8B161
      IF ( IBCX .EQ. 6 .AND. ITX .EQ. 1 ) CALL CHECK ('ITX   ')       SS8B162
      IF ( IBCY .EQ. 6 .AND. ITY .EQ. 1 ) CALL CHECK ('ITY   ')       SS8B163
      IF ( IERROR .EQ. 1 ) GO TO 99999                                SS8B164
      MWSIZ = NTWX*NTWY                                               SS8B165
      MUVSIZ = MATSIZ - MWSIZ                                         SS8B166
      MU = 0.                                                         SS8B167
```

```
      IF ( IBCY .EQ. 0 )  GO TO 8
      IF ( IKDF .EQ. 0 )  READ (5,5)  AA, BB, RR
      IF ( IKDF .EQ. 1 )  READ (5,5)  AA, BB, RR, MU
      GO TO 9
    8 IF ( IKDF .EQ. 0 )  READ (5,5)  AA, RR
      IF ( IKDF .EQ. 1 )  READ (5,5)  AA, RR, MU
      BB = 6.2831853 * RR
    9 CONTINUE
C **  THE BOUNDARY CONDITIONS ARE PRINTED
      II = IBCX
      IF (IBCY.NE.0) GO TO 20
      WRITE (6,10)
   10 FORMAT ('0THE BOUNDARY CONDITIONS OF THE COMPLETE CYLINDER AT X=0
     1AND X=A ARE')
      GO TO 40
   20 WRITE (6,150)
      IBCTAG = +1
      GO TO 40
   30 II=IBCY
      WRITE (6,160)
      IBCTAG = -1
   40 IF ( II-2 ) 70,80,50
   50 IF ( II-4 ) 90,100,60
   60 IF ( II-6 ) 110,120,130
   70 WRITE (6,170)
      GO TO 140
   80 WRITE (6,180)
      GO TO 140
   90 WRITE (6,190)
      GO TO 140
  100 WRITE (6,200)
      GO TO 140
  110 WRITE (6,210)
      GO TO 140
  120 WRITE (6,220)
      GO TO 140
  130 WRITE (6,230)
  140 IF (IBCTAG.GT.0.AND.IBCY.NE.0) GO TO 30
  150 FORMAT('0THE BOUNDARY CONDITIONS AT X=0 AND X=A ARE')
  160 FORMAT('0THE BOUNDARY CONDITIONS AT Y=0 AND Y=B ARE')
  170 FORMAT(' CLAMPED, SIMPLE')
  180 FORMAT(' SIMPLE, SIMPLE')
  190 FORMAT(' CLAMPED, CLAMPED')
  200 FORMAT(' CLAMPED, FREE')
  210 FORMAT(' SIMPLE, FREE')
  220 FORMAT(' FREE, FREE')
  230 FORMAT(' ELASTIC RESTRAINT')
      WRITE (6,240) NTUX,ITX,NTUY,ITY,MATSIZ,MATSIZ
  240 FORMAT ('0THERE ARE' ,I3,' MODES IN THE X DIRECTION, STARTING WITH
     1 M =',I3,' .'/' THERE ARE',I3,' MODES IN THE Y DIRECTION, STARTING
     2 WITH N =',I3,' .' / '0THE STIFFNESS MATRIX SIZE IS' I4,' BY'I4)
      IF (IFLAGD.NE.0) WRITE (6,250)
  250 FORMAT( '0A DYNAMIC SOLUTION WILL BE SOUGHT')
      IF (IFLAGB.NE.0) WRITE (6,260)
  260 FORMAT( '0A STABILITY SOLUTION WILL BE SOUGHT')
      IF (IFLAGW.NE.0) WRITE (6,270)
```

```
  270 FORMAT('0A SOLUTION UNDER LATERAL LOADS WILL BE SOUGHT')
      WRITE (6,280)  AA, BB, RR, MU
  280 FORMAT ('0A ='F20.5/'0B ='F20.5/'0R ='F20.5/'0MU ='F19.5)
C **  ELASTIC RESTRAINT
      IELAST = 1
      ALFAX = 0.
      BETAX = 0.
      ALFAY = 0.
      BETAY = 0.
      IF ( IBCX.EQ.7.AND.IBCY.LT.7 ) GO TO 290
      IF ( IBCX.EQ.7.AND.IBCY.EQ.7 ) GO TO 300
      IF ( IBCX.NE.7.AND.IBCY.EQ.8 ) GO TO 310
      IF ( IBCX.EQ.7.AND.IBCY.EQ.8 ) GO TO 320
      GO TO 330
  290 IELAST = 2
      READ (5,5) ALFAX,BETAX
      GO TO 330
  300 IELAST = 3
      READ (5,5) ALFAX,BETAX
      ALFAY = ALFAX
      BETAY = BETAX
      GO TO 330
  310 IELAST = 4
      READ (5,5) ALFAY,BETAY
      GO TO 330
  320 IELAST = 5
      READ (5,5) ALFAX,BETAX,ALFAY,BETAY
  330 CONTINUE
      IF (IELAST.EQ.1) GO TO 350
      WRITE (6,340) ALFAX,BETAX,ALFAY,BETAY
  340 FORMAT ('0THE ELASTIC RESTRAINT QUANTITIES ARE -- ' / ' ALFAX = '
     1 E16.8 / ' BETAX = ' E16.8 / ' ALFAY = ' E16.8 / ' BETAY = 'E16.8)
  350 CONTINUE
C **  READ IN NECESSARY MATERIAL PROPERTIES THROUGH STATEMENT 470
      DO 360 I=1,3
      DO 360 J=1,3
      AMAT(I,J)=0.
      BMAT(I,J)=0.
      DMAT(I,J)=0.
  360 CONTINUE
      IF ( IMATL . EQ . 1 )  GO TO 370
      IF ( IMATL . EQ . 2 )  GO  TO  390
      IF ( IMATL . EQ . 3 )  GO  TO  450
C **  SANDWICH
      DO 361 J=1,3,2
      READ (5,5) E1(J), E2(J), G(J), XNU12(J), H(J)
      THETA(J) = 0.
      IF ( IOUT .LT. 7 )  GO TO 361
      READ (5,5) (EC(I,J),I=1,3), (ET(I,J),I=1,3), XCHK
      MCHK(J) = XCHK + .1
      NCHK = MCHK(J)
      IF ( NCHK .LT. 1 .OR.NCHK .GT. 10 )  CALL CHECK ('MCHK  ')
      IF ( IERROR .EQ. 1 )  GO TO 99999
      READ(5,5) ( ANGCK(J,I), I=1,NCHK )
  361 CONTINUE
      READ (5,5) H(2)
```

```
      E1(2) = 1.
      E2(2) = 1.
      G(2)  = 1.
      XNU12(2) = .25
      THETA(2) = 0.
      DO 362 J=1,3,2
      NCHK = MCHK(J)
      IF ( J.EQ.1 ) K=KIN
      IF ( J.EQ.3 ) K=KOUT
      IF ( IOUT .LT. 7 ) WRITE(6,366) K,E1(J),E2(J),G(J),XNU12(J),H(J)
  366 FORMAT (/'OFOR THE ',A3,'ER FACING OF THE SANDWICH,  E1 =' ,E14.6
     1,', E2 =',E14.6,', G =',E14.6,', NU12 =',F7.3,', H =',F8.3)
      IF ( IOUT .GE. 7 )
     1WRITE (6,363) K, E1(J), E2(J), G(J), XNU12(J), H(J),
     1              ( EC(I,J), I=1,3) , ( ET(I,J), I=1,3),
     2              ( ANGCK(J,I), I=1,NCHK )
  363 FORMAT (/'OFOR THE ',A3,'ER FACING OF THE SANDWICH,  E1 =',
     1        E14.6,', E2 =',E14.6,', G =',E14.6,', NU12 =',F7.3,
     2        ', H =',F8.3//9X,'THE COMPRESSION ALLOWABLES IN THE 1, 2,
     3AND 12 DIRECTIONS ARE',3E15.6,        ' IN./IN.'//9X,
     4        'THE TENSION    ALLOWABLES IN THE 1, 2, AND 12 DIRECTIONS
     5 ARE',3E15.6,        ' IN./IN.'//9X,'THE ORIENTATIONS TO BE CHECKED
     6 ARE',10F8.2)
  362 CONTINUE
      WRITE (6,364) H(2)
  364 FORMAT ('OTHE CORE THICKNESS IS',F9.3,' IN.')
      T = H(1) + H(2) + H(3)
      WRITE (6,365) T
  365 FORMAT ('OTHE TOTAL SANDWICH THICKNESS IS',F9.3,' IN.')
      GO TO 410
C **  ISOTROPIC  - -  READ E, NU, AND T
  370 READ (5,5)  E1(1), XNU12(1), T
      IF ( IOUT .GE. 7 )  READ(5,5) (EC(I,1),I=1,3), (ET(I,1),I=1,3)
      WRITE (6,380)  E1(1),XNU12(1),T
  380 FORMAT ('OFOR THE ISOTROPIC MATERIAL, E ='E16.7,', NU ='F7.4,
     1        ', T = 'F9.4 )
      IF ( IOUT .GE. 7 )WRITE(6,381)(EC(I,1),I=1,3), (ET(I,1),I=1,3)
  381 FORMAT ('0',8X,'THE COMPRESSION ALLOWABLES IN THE 1, 2, AND 12 DIR
     1ECTIONS ARE',3E15.6,' IN./IN.'//9X,'THE TENSION ALLOWABLES IN THE
     21, 2, AND 12 DIRECTIONS ARE',3E15.6,' IN./IN.')
      AMAT(1,1) = E1(1)*T/(1.-XNU12(1)*XNU12(1))
      AMAT(2,2) = AMAT(1,1)
      AMAT(2,1) = XNU12(1)*AMAT(1,1)
      AMAT(1,2) = AMAT(2,1)
      AMAT(3,3) = E1(1)*T/2./(1.+XNU12(1))
      DMAT(1,1) = E1(1)*T*T*T/12./(1.-XNU12(1)*XNU12(1))
      DMAT(2,2) = DMAT(1,1)
      DMAT(2,1) = XNU12(1)*DMAT(1,1)
      DMAT(1,2) = DMAT(2,1)
      DMAT(3,3) = E1(1)*T*T*T/24./(1.+XNU12(1))
      E2(1) = E1(1)
      H(1) = T
      G(1) = E1(1)/2./(1+XNU12(1))
      THETA(1) = 0.
      GO TO 410
C **  LAMINATE WITH CONSTANT PLY PROPERTIES
```

```
390 READ (5,5) E1(1), E2(1), G(1), XNU12(1), H(1),
   1            ( THETA(I), I=1,NPLYS )
    IF ( IOUT .GE. 7 )  READ (5,5)   (EC(I,1),I=1,3), (ET(I,1),I=1,3)
    T = H(1)*NPLYS
    WRITE (6,400) NPLYS, E1(1), E2(1),         G(1), XNU12(1), H(1),
   1                T, ( THETA(I), I=1,NPLYS )
400 FORMAT ('OFOR THE ',I2,' PLY LAMINATE'/'OE1 =' E20.6 /'OE2 =',
   1         E20.6 / 'OG =' E20.6 / 'ONU12 = ' F6.4  / 'OH(I) =' F9.4/
   2         'OT = ' F9.4 / 'OTHE ORIENTATIONS ARE'/(' 'F10.4/))
    IF ( IOUT .GE. 7 )  WRITE(6,381) (EC(I,1),I=1,3), (ET(I,1),I=1,3)
    DO 409 I=1,NPLYS
    E1(I) = E1(1)
    E2(I) = E2(1)
    G(I)  = G(1)
    XNU12(I) = XNU12(1)
    H(I)  = H(1)
    DO 409 J=1,3
    EC(J,I) = EC(J,1)
    ET(J,I) = ET(J,1)
409 CONTINUE
410 CALL STIFF
420 WRITE (6,430) ((AMAT(I,J),J=1,3),(BMAT(I,J),J=1,3),I=1,3)
    WRITE (6,440) ((BMAT(I,J),J=1,3),(DMAT(I,J),J=1,3),I=1,3)
430 FORMAT ('1THE CONSTITUTIVE MATRIX IS'/ / / (6E16.7))
440 FORMAT (6E16.7)
C **  FIX FOR ELASTIC RESTRAINT
    IF ( IELAST .EQ. 1 )  GO TO 431
    ALFAX = ALFAX * AA / DMAT(1,1)
    BETAX = BETAX * AA / DMAT(1,1)
    ALFAY = ALFAY * BB / DMAT(2,2)
    BETAY = BETAY * BB / DMAT(2,2)
431 CONTINUE
    IF ( IMATL .EQ. 1 )  GO TO 470
    DO 601 I=1,3
    DO 601 J=1,3
601 A(I,J) = AMAT(I,J)
    DET = A(1,1)*A(2,2)*A(3,3) + A(1,2)*A(2,3)*A(3,1)
   1     + A(1,3)*A(2,1)*A(3,2) - A(1,3)*A(2,2)*A(3,1)
   2     - A(1,1)*A(2,3)*A(3,2) - A(1,2)*A(2,1)*A(3,3)
    AI(1,1) = ( A(2,2)*A(3,3) - A(2,3)*A(3,2) ) / DET
    AI(1,2) = ( A(2,3)*A(3,1) - A(2,1)*A(3,3)) / DET
    AI(1,3) = ( A(2,1)*A(3,2) - A(2,2)*A(3,1)) / DET
    AI(2,2) = ( A(1,1)*A(3,3) - A(1,3)*A(3,1) ) / DET
    AI(2,3) = ( A(1,2)*A(3,1) - A(1,1)*A(3,2)) / DET
    AI(3,3) = ( A(1,1)*A(2,2) - A(1,2)*A(2,1)) / DET
    EX = 1. / AI(1,1) / T
    EY = 1. / AI(2,2) / T
    GXY = 1. / AI(3,3) / T
    XNUXY = - AI(1,2) / AI(1,1)
    XNUYX = - AI(1,2) / AI(2,2)
    WRITE(6,441)
441 FORMAT ('OTHE LAMINATE PROPERTIES ARE')
    WRITE (6,442) EX,EY,GXY,XNUXY,XNUYX
442 FORMAT ('OEX =',E15.6,3X,'EY =',E15.6,3X,'G =',E15.6,3X,'NUXY =',
   1         F8.4,3X,'NUYX =',F8.4)
    GO TO 470
```

```
C **  LAMINATE WITH VARIABLE PLY PROPERTIES
  450 IF ( IOUT .LT. 7 )
     1READ (5,5) ( E1(I), E2(I), G(I), XNU12(I),H(I),THETA(I),I=1,NPLYS)
      IF ( IOUT .GE. 7 )  READ (5,5) ( E1(I), E2(I), G(I), XNU12(I),
     1    H(I), THETA(I), (EC(J,I),J=1,3), (ET(J,I),J=1,3), I=1,NPLYS )
      T = 0.
      DO 461 I=1,NPLYS
      WRITE(6,460) I,H(I),E1(I),E2(I),XNU12(I),G(I), THETA(I)
  460 FORMAT('OPLY'I4,' HAS A THICKNESS OF 'F11.7,' E1='E16.7,' E2='E16.
     17/'            NU12='F6.4, ' G='E16.7,' AND ORIENTATION=' F10.3,' DEGR
     2EES.')
      IF ( IOUT .GE. 7 ) WRITE(6,381)(EC(J,I),J=1,3),(ET(J,I),J=1,3)
  461 T = T + H(I)
      WRITE (6,462)  T
  462 FORMAT ('OT = ',F9.4,' IN.')
      GO TO 410
  470 CONTINUE
      IF ( NSTRNG .EQ. 0 )  GO TO 490
C **  FOR STRINGERS
      IF ( IEQS .EQ. 1 )  GO TO 471
      READ (5,5) ( YSTRNG(L),YBARS(L), ZBARS(L), AS(L), XIYYS(L),
     1               XIYZS(L), XIZZS(L), ES(L), GJS(L), RHOS(L),
     2                          L=1,NSTRNG )
      WRITE (6,477)
  477 FORMAT ('OTHE STRINGER PROPERTIES FOLLOW  --')
      WRITE(6,479)
  479 FORMAT ('0',T2,'L',T9,'Y',T16,'YBAR',T23,'ZBAR',T30,'AREA',
     1         T40,'IYY',T52,'IYZ',T64,'IZZ',T77,'E',T88,'GJ',T100,
     2         'RHO'/)
      WRITE(6,480)(L,YSTRNG(L),YBARS(L), ZBARS(L), AS(L), XIYYS(L),
     1               XIYZS(L), XIZZS(L), ES(L), GJS(L), RHOS(L),
     2                          L=1,NSTRNG )
  480 FORMAT (1X,0PI3,F9.2,3F7.2,1P6E12.4)
      GO TO 489
  471 READ (5,5)    YBARS(1), ZBARS(1), AS(1), XIYYS(1), XIYZS(1),
     1               XIZZS(1), ES(1), GJS(1), RHOS(1)
      YSTRNG(1) = BB/(NSTRNG + 1)
      IF ( IBCY .EQ. 0 )  YSTRNG(1) = BB/NSTRNG
      DO 472 L=2,NSTRNG
      YSTRNG(L) = L * YSTRNG(1)
      YBARS(L) = YBARS(1)
      ZBARS(L) = ZBARS(1)
      AS(L)    = AS(1)
      XIYYS(L) = XIYYS(1)
      XIYZS(L) = XIYZS(1)
      XIZZS(L) = XIZZS(1)
      ES(L)    = ES(1)
      GJS(L)   = GJS(1)
  472 RHOS(L)  = RHOS(1)
      WRITE (6,473) NSTRNG
  473 FORMAT ('OTHERE ARE ',I3,' EQUALLY SPACED STRINGERS EACH OF WHICH
     1HAS THE FOLLOWING PROPERTIES --')
      WRITE (6,474)
  474 FORMAT ('0',T6,'SPACING ',T16,'YBAR',T23,'ZBAR',T30,'AREA',
     1         T40,'IYY',T52,'IYZ',T64,'IZZ',T77,'E',T88,'GJ',T100,
     2         'RHO'/)
```

```
      WRITE (6,475) YSTRNG(1), YBARS(1), ZBARS(1), AS(1), XIYYS(1),
     1              XIYZS(1), XIZZS(1), ES(1), GJS(1), RHOS(1)
  475 FORMAT (4X,F9.2,3F7.2,1P6E12.4)
  489 DO 476 L=1,NSTRNG
  476 YSTRNG(L) = YSTRNG(L)/BB
  490 CONTINUE
      IF ( NRING .EQ. 0 )  GO TO 510
C **  FOR RINGS
      IF ( IEQR .EQ. 1 )  GO TO 501
      READ (5,5)  ( XRINGS(K), XBARR(K), ZBARR(K), AR(K), XIXXR(K),
     1              XIXZR(K), XIZZR(K), ER(K), GJR(K), RHOR(K),
     2                        K=1,NRING )
      WRITE (6,498)
  498 FORMAT ('OTHE RING PROPERTIES FOLLOW --')
      WRITE (6,500)
  500 FORMAT ('0',T2,'K',T9,'X',T16,'XBAR',T23,'ZBAR',T30,'AREA',
     1        T40,'IXX',T52,'IXZ',T64,'IZZ',T77,'E',T88,'GJ',T100,
     2        'RHO'/)
      WRITE(6,480)(K,XRINGS(K), XBARR(K), ZBARR(K), AR(K), XIXXR(K),
     1             XIXZR(K), XIZZR(K), ER(K), GJR(K), RHOR(K),
     2                        K=1,NRING )
      GO TO 509
  501 READ (5,5)   XBARR(1), ZBARR(1), AR(1), XIXXR(1), XIXZR(1),
     1             XIZZR(1), ER(1), GJR(1), RHOR(1)
      XRINGS(1) = AA/(NRING + 1)
      DO 502 K=2,NRING
      XRINGS(K) = K * XRINGS(1)
      XBARR(K) = XBARR(1)
      ZBARR(K) = ZBARR(1)
      AR(K)    = AR(1)
      XIXXR(K) = XIXXR(1)
      XIXZR(K) = XIXZR(1)
      XIZZR(K) = XIZZR(1)
      ER(K)    = ER(1)
      GJR(K)   =  GJR(1)
  502 RHOR(K)  =  RHOR(1)
      WRITE (6,503) NRING
  503 FORMAT ('OTHERE ARE ',I2,' EQUALLY SPACED RINGS EACH OF WHICH HAS
     1THE FOLLOWING PROPERTIES --')
      WRITE (6,504)
  504 FORMAT ('0',T6,'SPACING ',T16,'XBAR',T23,'ZBAR',T30,'AREA',
     1        T40,'IXX',T52,'IXZ',T64,'IZZ',T77,'E',T88,'GJ',T100,
     2        'RHO'/)
      WRITE (6,475) XRINGS(1), XBARR(1), ZBARR(1), AR(1), XIXXR(1),
     1              XIXZR(1), XIZZR(1), ER(1), GJR(1), RHOR(1)
  509 DO 505 K=1,NRING
  505 XRINGS(K) = XRINGS(K) / AA
  510 CONTINUE
      IF ( IFLAGD .EQ. 0 )  GO TO 550
C **  IF DOING DYNAMICS, READ AVERAGE MATERIAL DENSITY
      READ (5,5) DENSE
      WRITE (6,520) DENSE
  520 FORMAT ('OTHE MATERIAL DENSITY = 'E15.8,' LB.-SEC.**2/IN.**4')
      RHAB = DENSE * T * AA * BB
      IF ( NLMASS .EQ. 0 )  GO TO 550
C **  HAVE LUMPED MASSES
```

```
      DO 530 I=1,NLMASS
      READ (5,5)  X, Y, PMASS(I)
      IPWW(I) = X + .1
      IPWY(I) = Y + .1
  530 WRITE (6,540) IPWW(I), IPWY(I), PMASS(I)
  540 FORMAT ('OTHERE IS A LUMPED MASS AT COORDINATES'I3,',',I3,
     1        ' OF MAGNITUDE'E15.7,' LB-SEC**2/IN')
  550 CONTINUE
      IF ( IEDGE .EQ. 0 )  GO TO 610
C **  READ EDGE LOADS
      IF ( IBCY .EQ. 0 .AND. IEDGE .EQ. 2 )  CALL CYLNDR ( &571 )
      IF ( NPNX .GT. 0 .AND. NPNY .GT. 0 )  GO TO 570
      WRITE (6,560)
  560 FORMAT ('1EDGE LOADS ARE TO BE INCLUDED BUT NPNX OR NPNY IS ZERO.'
     1       /' THIS PROBLEM IS TERMINATED.')
      GO TO 99999
  570 READ (5,5) (( PX(J,I), PY(J,I), PXY(J,I), J=1,NPNX ), I=1,NPNY )
  571 CONTINUE
      IEDGE = 1
      WRITE (6,580) (( PX(J,I), J=1,NPNX ), I=1,NPNY )
  580 FORMAT ('OPX(I,J) FOLLOWS'/(1P10E12.4))
      WRITE (6,590) (( PY(J,I), J=1,NPNX ), I=1,NPNY )
  590 FORMAT ('OPY(I,J) FOLLOWS'/(1P10E12.4))
      WRITE (6,600) (( PXY(J,I), J=1,NPNX ), I=1,NPNY )
  600 FORMAT ('OPXY(I,J) FOLLOWS'/(1P10E12.4))
      IF ( NSTRNG + NRING .EQ. 0 )  GO TO 610
      IF ( NSTRNG .EQ. 0 )  GO TO 608
      DO 604 L=1,NSTRNG
      V(1,1) = 1.
      Y = YSTRNG(L)
      DO 602 K=2,NPNY
  602 V(1,K) = Y ** (K-1)
      XNX = 0.
      DO 603 K=1,NPNY
  603 XNX = XNX + PX(1,K) * V(1,K)
  604 PAXS(L) = XNX * AS(L) * ES(L) / EX / T
      IF ( NPNY .EQ. 1 .AND. IEQS .EQ. 1 )  WRITE (6,605) PAXS(1)
  605 FORMAT ('OTHE AXIAL LOAD CARRIED BY EACH STRINGER IS ',E12.5,' LBS
     1.')
      IF ( NPNY .NE. 1 .OR. IEQS .EQ. 0 ) WRITE (6,606)
  606 FORMAT ('OTHE AXIAL LOADS ( LBS. ) CARRIED BY THE STRINGERS FOLLOW
     1 --')
      IF(NPNY.NE.1.OR.IEQS.EQ.0) WRITE(6,607)(L,PAXS(L),L=1,NSTRNG)
  607 FORMAT ('0',8(I3,E13.5))
  608 IF ( NRING .EQ. 0 ) GO TO 610
      IF ( IBCY .EQ. 0 ) GO TO 610
      V(1,1) = 1.
      DO 612 K=1,NRING
      X = XRINGS(K)
      DO 609 L=2,NPNX
  609 V(1,L) = X ** (L-1)
      XNY = 0.
      DO 611 L=1,NPNX
  611 XNY = XNY + PY(L,1) * V(1,L)
  612 PAXR(K) = XNY * AR(K) * ER(K) / EY / T
      IF ( NPNX .EQ. 1 .AND. IEQR .EQ. 1 ) WRITE (6,613) PAXR(1)
```

```
  613 FORMAT ('OTHE LOAD CARRIED BY EACH RING IS ',E12.5,' LBS.')
      IF ( NPNX .NE. 1 .OR. IEQR .EQ. 0 ) WRITE (6,614)
  614 FORMAT ('OTHE LOADS ( LBS. ) CARRIED BY THE RINGS FOLLOW --')
      IF(NPNX.NE.1.OR.IEQR.EQ.0) WRITE(6,607)(K,PAXR(K),K=1,NRING)
  610 CONTINUE
      IF ( IFLAGW .NE. 1 )  GO TO 650
C **  READ LATERAL LOADS
      IF ( NQTX .GT. 0 .AND. NQTY .GT. 0 )  GO TO 630
      WRITE (6,620)
  620 FORMAT ('1LATERAL LOADS ARE TO BE INCLUDED BUT NQTX OR NQTY IS ZER
     1O.'     /' THIS PROBLEM IS TERMINATED.')
      GO TO 99999
  630 READ (5,5) (( Q(J,I), J=1,NQTX ), I=1,NQTY )
      WRITE(6,640)((Q(J,I), J=1,NQTX ), I=1,NQTY )
  640 FORMAT ('OQ(I,J) FOLLOWS' / (1P10E12.4))
  650 CONTINUE
      IF ( NPTLDS .EQ. 0 )  GO TO 680
C **  HAVE POINT LOADS
      DO 660 I=1,NPTLDS
      READ (5,5) X, Y, DUM
      IPXX(I) = X + .1
      IPYY(I) = Y + .1
      PC(I) = DUM
  660 WRITE (6,670)  IPXX(I), IPYY(I), PC(I)
  670 FORMAT ('OTHERE IS A CONCENTRATED LOAD AT COORDINATES'I3,',' I3,
     1         ' OF MAGNITUDE'F12.5,' LBS.')
  680 CONTINUE
      IF ( NPTMOM .EQ. 0 )  GO TO 710
C **  HAVE POINT MOMENTS
      DO 690 I=1,NPTMOM
      READ (5,5) X, Y, TAG, DUM
      IFXX(I) = X + .1
      IFYY(I) = Y + .1
      FC(I) = DUM
      ITAGCM(I) =  TAG + .1
      DIR = XDIR
      IF ( ITAGCM(I) .EQ. 2 )  DIR = YDIR
  690 WRITE (6,700)  DIR, IFXX(I), IFYY(I), FC(I)
  700 FORMAT ('OTHERE IS A CONCENTRATED MOMENT ABOUT THE ',A1,' AXIS AT
     1COORDINATES'I3,',',I3,' OF MAGNITUDE'E15.7,' IN.-LBS.')
  710 CONTINUE
      IF ( NLNMOM .EQ. 0 )  GO TO 750
C **  HAVE LINE MOMENTS
      DO 730 I=1,NLNMOM
      READ (5,5) TAG, DIST, PLMOM(I)
      ITAGLM(I) = TAG + .1
      IDISLM(I) = DIST + .1
      IF ( ITAGLM(I) .EQ. 2 )  GO TO 720
      DIR = XDIR
      GO TO 730
  720 DIR = YDIR
  730 WRITE (6,740)       DIR, IDISLM(I), PLMOM(I)
  740 FORMAT ( 'OTHERE IS A LINE MOMENT PARALLEL TO THE ',A1,' AXIS ON G
     1RID LINE ',    I2,' WITH A MAGNITUDE OF ',E15.7,' IN-LB/IN')
  750 CONTINUE
      IF ( NPTSUP .EQ. 0 )  GO TO 780
```

```
C **  HAVE POINT SPRINGS SPECIFIED AT GRID POINTS
     DO 760 I=1,NPTSUP
     READ (5,5)  X, Y, PKC(I)
     IGSPRX(I) = X + .1
     IGSPRY(I) = Y + .1
 760 WRITE (6,770)  IGSPRX(I), IGSPRY(I), PKC(I)
 770 FORMAT('OTHERE IS AN ELASTIC SUPPORT AT COORDINATES',I3,',',I3,
    1        ' WITH A SPRING CONSTANT OF'E16.8,' LB/IN.')
 780 CONTINUE
     IF ( NLNSPR .EQ. 0 )  GO TO 830
C **  HAVE LINE SPRINGS
     DO 810 I=1,NLNSPR
     READ (5,5)   TAG, DIST, PLINE(I)
     ITAGLS(I) = TAG + .1
     IDISLS(I) = DIST + .1
     DIR = XDIR
     IF ( ITAGLS(I) .EQ. 2 )  DIR = YDIR
 810 WRITE (6,820) DIR, IDISLS(I), PLINE(I)
 820 FORMAT ( 'OTHERE IS A LINE SPRING PARALLEL TO THE ',A1,' AXIS ON G
    1RID LINE ',I2,' WITH A SPRING CONSTANT OF ',E15.7,' LB/IN/IN.')
 830 CONTINUE
     IF ( IPRTN + IPRTQ .EQ. 0 )  GO TO 950
     DO 890 I=1,5
     X = .25 * (I-1)
     V(1,1) = 1.
     DO 840 K=2,10
 840 V(1,K) = X ** (K-1)
     DO 890 J=1,5
     Y = .25 * (J-1)
     V(2,1) = 1.
     DO 850 K=2,10
 850 V(2,K) = Y ** (K-1)
     PRTNX(I,J) = 0.
     PRTNY(I,J) = 0.
     PRTNXY(I,J) = 0.
     PRTQ(I,J) = 0.
     IF ( IPRTN .EQ. 0 )  GO TO 870
     DO 860 K=1,NPNX
     DO 860 L=1,NPNY
     PRTNX (I,J) = PRTNX (I,J) + PX (K,L) * V(1,K) * V(2,L)
     PRTNY (I,J) = PRTNY (I,J) + PY (K,L) * V(1,K) * V(2,L)
 860 PRTNXY(I,J) = PRTNXY(I,J) + PXY(K,L) * V(1,K) * V(2,L)
 870 IF ( IPRTQ .EQ. 0 )  GO TO 890
     DO 880 K=1,NQTX
     DO 880 L=1,NQTY
 880 PRTQ(I,J) = PRTQ(I,J) + Q(K,L) * V(1,K) * V(2,L)
 890 CONTINUE
     IF ( IPRTN .EQ. 0 )  GO TO 930
     WRITE (6,900)
 900 FORMAT ('1NX, NY, AND NXY, RESPECTIVELY, ARE PRINTED AT QUARTER PO
    1INTS OF THE PANEL'//)
     WRITE (6,910) (( PRTNX(I,J), J=1,5), I=1,5)
 910 FORMAT (' ',5E20.7)
     WRITE (6,920)
 920 FORMAT ('0')
     WRITE (6,910) (( PRTNY(I,J), J=1,5), I=1,5)
```

```
      WRITE (6,920)                                                     SS8B672
      WRITE (6,910) (( PRTNXY(I,J), J=1,5), I=1,5)                      SS8B673
  930 IF ( IPRTQ .EQ. 0 )  GO TO 950                                    SS8B674
      WRITE (6,940)                                                     SS8B675
  940 FORMAT ('1THE LATERAL LOAD DISTRIBUTION IS PRINTED AT QUARTER POINSS8B676
     1TS OF THE PANEL'//)                                               SS8B677
      WRITE (6,910) (( PRTQ(I,J), J=1,5), I=1,5)                        SS8B678
  950 CONTINUE                                                          SS8B679
      IF ( IFLEX .EQ. 0 )  GO TO 970                                    SS8B680
      READ (5,5) ( XP(I), YP(I), I=1,IFLEX )                            SS8B681
      WRITE (6,960)  IFLEX, ( XP(I), YP(I), I=1,IFLEX )                 SS8B682
  960 FORMAT ('0THE',I3,' NORMALIZED POINTS FOR THE FLEXIBILITY MATRIX ASS8B683
     1RE'//3(6X,1HX,10X,1HY,4X)/(/6F11.5))                              SS8B684
  970 CONTINUE                                                          SS8B685
 9999 RETURN                                                            SS8B686
99999 CALL SKIPPR                                                       SS8B687
      GO TO 1                                                           SS8B688
      END                                                               SS8B689

                                                              CC = 00690
```

```
      SUBROUTINE  CYLNDR  ( * )
C
C **  THIS SUBROUTINE CALCULATES THE NX, NY, AND NXY VALUES TO BE
C **  USED WHEN A SHELL IS LOADED BY AN AXIAL FORCE, A TORQUE,
C **  AND/OR A BENDING MOMENT.
C
      DIMENSION    PX(10,10),     PY(10,10),     PXY(10,10),    V(10)
C
      COMMON / GEOM   / AA,        BB,        RR
      COMMON / NUMBER / NNUM(10), NPNX,      NPNY
      COMMON / PARAM  / PDUM(1650),          PX,        PY,        PXY
C
      DATA V(1) / 1.00225 /,  V(2) / .140605 /,  V(3) / -23.2379 /
      DATA V(4) / 19.8787 /,  V(5) / 28.8562 /,  V(6) / -3.39401 /
      DATA V(7) /-25.2977 /,  V(8) /-15.1520 /,  V(9) /  16.6307 /
      DATA V(10)/ 1.57479 /
C
      TORQUE = 0.
      PI = 3.1415926536
      NPNX = 1
    5 FORMAT (1X)
      READ (5,5)  FAXIAL,          BNDMOM
      WRITE (6,6) FAXIAL, TORQUE, BNDMOM
    6 FORMAT ('0THE APPLIED CYLINDER LOADS ARE --'/
     1       ' ',T40,'AXIAL FORCE      =',E15.6,T74,'LBS.'/
     2       ' ',T40,'TORQUE           =',E15.6,T74,'IN-LBS.'/
     3       ' ',T40,'BENDING MOMEMT   ='E15.6,T74,'IN-LBS.')
      PF = FAXIAL / 2. / PI / RR
      PT = TORQUE / 2. / PI / RR / RR
      PM = BNDMOM / PI / RR / RR
      IF (BNDMOM .GT. .0001 )  GO TO 10
      NPNY = 1
      PX (1,1) = PF
      PY (1,1) = 0.
      PXY(1,1) = PT
      RETURN 1
   10 NPNY = 10
      DO 80 J=1,10
      PX(1,J) = V(J)*PM
      PY(1,J) = 0.
   80 PXY(1,J)= 0.
      PX(1,1) = PX(1,1) + PF
      PXY(1,1)= PT
      RETURN 1
      END
```

CC = 00045

```
      SUBROUTINE CHECK (A)                                              SS8D000
      REAL*8 A                                                          SS8D001
      COMMON / CHECKS / IERROR                                          SS8D002
      IERROR = 1                                                        SS8D003
      WRITE (6,6) A                                                     SS8D004
    6 FORMAT ('OTHE PROGRAM HAS READ AN UNACCEPTABLE VALUE FOR ',A6 /   SS8D005
     1       ' THE NEXT PROBLEM WILL BE ATTEMPTED AFTER CHECKING THE COSS8D006
     2NTROL VARIABLES')                                                 SS8D007
      RETURN                                                            SS8D008
      END                                                               SS8D009
```

CC = 00010

```
      SUBROUTINE  STIFF
C     THIS SUBROUTINE CALCULATES THE 6 BY 6 ARRAY OF STIFFNESS TERMS AT
C     A POINT FOR A LAMINATED PLATE.  THE INPUT IS THE NUMBER OF PLIES
C     (MPLY), THE ORIENTATIONS OF THE PLIES (TETA( )), THE THICKNESS OF
C     EACH PLY (THICK), AND THE MATERIAL PROPERTIES OF THE ORTHOTROPIC
C     PLIES (E1,E2,G, AND POISSON'S RATIO (U1)).
C **  REVISED FOR CURVED PANELS  - -  8/69
      DIMENSION AH(41), CB(3,3,40)
      DIMENSION    C1(40), C2(40), C3(40), C11(40), C22(40), C12(40)
      COMMON / ABD /    A(3,3),  DS(3,3),  DP(3,3),  RHAB,  TETA(40),
     1                  THICK(40),  E1(40),  E2(40),  G(40),  U1(40),
     2                   EC(3,40), ET(3,40), ANGCK(3,10), MCHK(3), AH
      COMMON / NUMBER / MPLY
      COMMON / CNTROL / IDUM(5),  IMATL
      EQUIVALENCE  (C1(1),E1(1)),(C2(1),E2(1)),(C3(1),U1(1))
C     THE MIDDLE SURFACE IS LOCATED
      MPLY2= MPLY+1
      AHK=0.
      DO 100 I=1, MPLY
  100 AHK= AHK + THICK(I)/2.
      AH(1)=-AHK
      DO 30 I=2, MPLY2
   30 AH(I)= AH(I-1)+ THICK(I-1)
C     THE CBAR ARRAY IS CALCULATED FOR EACH PLY, USING DOUBLE-ANGLE
C     TRANSFORMATION FORMULAS.
      DO 40 N=1, MPLY
      U2= U1(N)*E2(N)/E1(N)
      DEL= 1.-U2*U1(N)
      CC1= E1(N)/DEL
      CC2= E2(N)/DEL
      CC3= CC1*U2
      CC4= G(N)
      C11(N) = CC1
      C22(N) = CC2
      C12(N) = CC3
      IF ( IMATL .EQ. 1 )   GO TO 40
      COT = 2.*TETA(N)*.017453292519943
      CO2= COS(COT)
      CO4= COS(2.*COT)
      SN2= SIN(COT)
      SN4= SIN(2.*COT)
      AJ1= CC1+CC2+2.*CC3
      AJ2= CC4- CC3
      CB(1,1,N)=.375*AJ1+.5*AJ2+(CC1-CC2)/2.*CO2+(AJ1/8.+AJ2/2.-CC4)*CO4
      CB(1,2,N)=AJ1/8. -AJ2/2.+(CC4-AJ1/8.-AJ2/2.)*CO4
      CB(2,1,N)= CB(1,2,N)
      CB(1,3,N)=(CC1- CC2)/4.*SN2 +(AJ1/8.+AJ2/2.- CC4)*SN4
      CB(3,1,N)=CB(1,3,N)
      CB(2,2,N)= CB(1,1,N) +(CC2-CC1)*CO2
      CB(2,3,N)= CB(1,3,N) -(AJ1/4.+AJ2-CC4*2.)*SN4
      CB(3,2,N)= CB(2,3,N)
      CB(3,3,N)= AJ1/8. +AJ2/2. +(CC4 -AJ1/8.-AJ2/2.)*CO4
   40 CONTINUE
C     THE A, DSTAR, AND D MATRICES ARE CALCULATED AND STORED IN A( , ),
C     DS( , ), AND DP( , ).
      IF ( IMATL .EQ. 1 )   GO TO 51
```

```
      DO 50 I=1,3                                                       SS8E056
      DO 50 J=I,3                                                       SS8E057
      A(I,J)=0.                                                         SS8E058
      DS(I,J)=0.                                                        SS8E059
      DP(I,J)=0.                                                        SS8E060
      AX=AH(1)*AH(1)                                                    SS8E061
      DO 60 K=1,MPLY                                                    SS8E062
      A(I,J)=A(I,J)+ CB(I,J,K) *(AH(K+1)-AH(K))                         SS8E063
      AY=AX                                                             SS8E064
      AX=AH(K+1)*AH(K+1)                                                SS8E065
      DP(I,J)= DP(I,J)+ CB(I,J,K)*(AX*AH(K+1)-AY*AH(K))                 SS8E066
   60 DS(I,J)= DS(I,J)+ CB(I,J,K)*(AX-AY)                               SS8E067
      DP(I,J)=DP(I,J)/3.                                                SS8E068
      DS(I,J)= DS(I,J)/2.                                               SS8E069
      DP(J,I)= DP(I,J)                                                  SS8E070
      DS(J,I)= DS(I,J)                                                  SS8E071
   50 A(J,I)= A(I,J)                                                    SS8E072
   51 CONTINUE                                                          SS8E073
      DO 70 N=1,MPLY                                                    SS8E074
      C1(N) = C11(N)                                                    SS8E075
      C2(N) = C22(N)                                                    SS8E076
   70 C3(N) = C12(N)                                                    SS8E077
      RETURN                                                            SS8E078
      END                                                               SS8E079
                                                              CC = 00080
```

```
      SUBROUTINE TABLE
C
C **  THIS SUBROUTINE SERVES AS A CONTROL PROGRAM FOR THE CALCULATION
C **  OF THE TABLE OF INTEGRALS.
C
      DIMENSION    AL(1,2,6,3,10,3,10),      EVAL(4,2,3,10,25), TIME(50),
     1             $W(10,2,3,10,10),    P(11,2,3,3,10),         ITIME(12)
C
      COMMON / ARRAYS / P,        AL,        $W
      COMMON / VALUES / EVAL
      COMMON / NUMBER / N1,       NTUX,      NTVX,      NTWX,      NTUY,
     1                  NTVY,     NTWY,      NMODES,    NSTRNG,    NRING,
     2                  NPNX,     NPNY,      NQTX,      NQTY,      N$(9),
     3                  ITX,      ITY
      COMMON / CNTROL / N3(3),    IBCX,      IBCY,     N4(7),      INTPRT
      COMMON / GEOM   / ADUM(3),  ALFAX,     ALFAY,     BETAX,     BETAY
      COMMON / $TIME  / TIME,     ITIME
C
      CALL STATUS ( ITIME )
      TIME(3) = ITIME(8)/100.
      ET = TIME(3) - TIME(1)
      IF ( INTPRT .EQ. 1 )  WRITE (6,10)  ET
   10 FORMAT ('OELAPSED TIME BEFORE TABLE GENERATION = 'F7.3)
      MAX$X = MAXO ( NPNX, NQTX, 1 )
      MAX$Y = MAXO ( NPNY, NQTY, 1 )
      MAX$XY = MAXO ( MAX$X, MAX$Y )
      MAXP1 = MAX$XY + 1
C
      JBCX = IBCX
      JBCY = IBCY
      CALL INTEGL ( ALFAX, BETAX, JBCX, NTUX, MAX$X, 1, ITX )
      CALL INTEGL ( ALFAY, BETAY, JBCY, NTUY, MAX$Y, 2, ITY )
C
  190 CALL STATUS ( ITIME )
      TIME(4) = ITIME(8)/100.
      ET = TIME(4) - TIME(3)
      IF ( INTPRT .EQ. 1 )  WRITE (6,200)  ET
  200 FORMAT ('OINTEGRAL EVALUATION TIME = 'F7.3)
      RETURN
      END
```

CC = 00040

```
      SUBROUTINE INTEGL ($ALFA,$BETA,MNIJ,NTERMS,IPOWER,IDEFNE,IZ)       SS8G000
C     THIS SUBROUTINE COMPUTES AND RETURNS, WITH THE AID CF 'PPP',       SS8G001
C     'SPECAL', AND ELASTC, THE INTEGRALS AND MODE SHAPE EVALUATIONS FORSS8G002
C     ANY OF THE BEAM CONDITIONS CONSIDERED.  THE INPUT IS $ALFA, $BETA,SS8G003
C     AND MNIJ. $ALFA,$BETA ARE USED IN SUBROUTINE ELASTC IF AND ONLY   SS8G004
C     IF MNIJ IS GREATER THAN 6.  IF MNIJ IS LESS THAN 7, THE INITIAL   SS8G005
C     FREQUENCY ESTIMATES ARE READ INTO EP( ).  THESE ESTIMATES ARE USEDSS8G006
C     WITH A NEWTON-RAPHSON ITERATION ON THE APPROPRIATE FREQUENCY      SS8G007
C     EQUATION TO OBTAIN ACCURATE FREQUENCIES AND MODE SHAPES.  THE     SS8G008
C     RESULTS ARE RETURNED THROUGH THE COMMON BLOCK ARRAYS. THE ROUTINE SS8G009
C     IS IN DOUBLE PRECISION .                                          SS8G010
C **  REVISED FOR CURVED PANELS  - -  8/69                              SS8G011
      IMPLICIT REAL*8(A-H,O-Z), INTEGER  (I-N)                          SS8G012
      DIMENSION C(4,4,3,10),CLASTC(4,10),FFF(10)                        SS8G013
      COMMON / BLOCK /        AL(1,6,3,10,3,10),  EVAL(4,3,10,25),      SS8G014
     1              EVQ(4,3,2,25),       PZ(11,3,3,10),                 SS8G015
     2              TH(10,4,4,3),        ALVA(11,11,2),      P(11,10),  SS8G016
     3              CE(4,10), E(4,4),    EP(10),     CN(4),    CM(4)    SS8G017
      COMMON / ARRAYS / $P(11,2,3,3,10),  $AL(1,2,6,3,10,3,10),         SS8G018
     1                  $W(10,2,3,10,10)                                SS8G019
      COMMON / VALUES / $EVAL(4,2,3,10,25)                              SS8G020
      COMMON / NUMBER / NDUM(8),  NSTRNG,    NRING                      SS8G021
      COMMON / STFVAL / $ESV(10,100),   $ESW(10,100),   $ESDW(10,100),  SS8G022
     1                  $ERU(10,50),    $ERW(10,50),    $ERDW(10,50),   SS8G023
     2                  $STRNG(100),    $RINGS(50)                      SS8G024
      COMMON / CNTROL / IFLAGD, IFLAGB                                  SS8G025
C     MNIJ IS A FLAG FOR BOUNDARY CONDITION                             SS8G026
C     MNIJ = 0 FOR FULL CYLINDER                                        SS8G027
C     MNIJ=1 FOR FIXED SIMPLE BEAM, =2 FOR SIMPLE-SIMPLE, =3 FOR FIXED- SS8G028
C     FIXED, =4 FOR FIXED-FREE, =5 FOR SIMPLE FREE, AND = 6 FOR FREE-   SS8G029
C     FREE.  GREATER THAN 6 IS USED FOR ELASTICALLY RESTRAINED.         SS8G030
      PIE = 3.1415926535898                                             SS8G031
      S3 = DSQRT (3.D0)                                                 SS8G032
      IF(MNIJ .GT. 6) GO TO 700                                         SS8G033
C     ASH= 0 ,IJKLM=-1,IKJ=1  FOR A SIMPLE-SIMPLE BEAM                  SS8G034
C     ASH=-1.,IJKLM= 0,IKJ=1  FOR A FIXED-FIXED BEAM                    SS8G035
C     ASH=-1.,IJKLM=+1,IKJ=3  FOR A FREE-FREE BEAM                      SS8G036
C     ASH=+1.,IJKLM= 0,IKJ=1  FOR A FIXED-FREE BEAM                     SS8G037
C     ASH= 0 ,IJKLM=-2,IKJ=2  FOR A SIMPLE-FREE BEAM                    SS8G038
C     ASH= 0 ,IJKLM=-3,IKJ=1  FOR A FIXED-SIMPLE BEAM                   SS8G039
      ICYL=0                                                            SS8G040
      IF(MNIJ.NE.0) GO TO 2999                                          SS8G041
C     CYLINDER                                                          SS8G042
      ICYL=1                                                            SS8G043
      ASH = 0.                                                          SS8G044
      IJKLM = -1                                                        SS8G045
      IKJ = 1                                                           SS8G046
      I500 = 2                                                          SS8G047
      IF ( IFLAGB .NE. 0 )  GO TO 2997                                  SS8G048
      EP(1) = IZ * 6.28319                                              SS8G049
      DO 3000 I=2,NTERMS                                                SS8G050
 3000 EP(I) = EP(I-1) + 6.283                                           SS8G051
      GO TO 3009                                                        SS8G052
 2997 EP(1) = IZ * 3.14159                                              SS8G053
      DO 2998 I=2,NTERMS                                                SS8G054
 2998 EP(I) = EP(I-1) + 3.14159                                         SS8G055
```

```
      GO TO 3009
 2999 IF(MNIJ.NE.1) GO TO 3001
C     CLAMPED - SIMPLE
      ASH=0.
      IJKLM= -3
      IKJ = 1
      EP(1) = (4.*IZ + 1.) * PIE / 4.
      GO TO 3007
 3001 IF(MNIJ.NE.2)GO TO 3002
C     SIMPLE - SIMPLE
      ASH=0.
      IJKLM=-1
      IKJ=1
      EP(1) = IZ * 3.14159
      GO TO 3007
 3002 IF(MNIJ.NE.3)GO TO 3003
C     CLAMPED - CLAMPED
      ASH=-1.
      IJKLM=0
      IKJ=1
      EP(1) = (2.*IZ + 1.) * PIE / 2.
      GO TO 3007
 3003 IF(MNIJ.NE.4) GO TO 3004
C     CLAMPED - FREE
      ASH=1.
      IJKLM=0
      IKJ=1
      EP(1) = (2.*IZ - 1.) * PIE / 2.
      GO TO 3007
 3004 IF(MNIJ.NE.5)GO TO 3005
C     SIMPLE - FREE
      ASH=0.
      IJKLM=-2
      IF ( IZ .NE. 1 )  GO TO 3105
      IKJ= 2
      EP(1)= 3.
      EP(2)= 3.93
      GO TO 3007
 3105 IKJ = 1
      EP(1) = ( 4.*IZ - 3. ) * PIE / 4.
      GO TO 3007
 3005 ASH= -1.
C     FREE - FREE
      IJKLM= 1
      IF ( IZ .NE. 0 )  GO TO 3100
      IKJ=3
      EP(1)=3.
      EP(2)=2.
      EP(3)=4.73
      GO TO 3007
 3100 IKJ = 1
      EP(1) = ( 2.*IZ - 1. ) * PIE / 2.
 3007 I500=IKJ+1
      DO 3008 I=I500,NTERMS
 3008 EP(I) = EP(I-1)+3.142
C     COMPUTE ACCURATE FREQUENCIES FROM HERE TO 200
```

```
 3009 CONTINUE
      DO 200 I=IKJ,NTERMS
      DO 200 J=1,8
      DC=DCOS(EP(I))
      DS=DSIN(EP(I))
      DX=DEXP(EP(I))
      DCH=.5*(DX+1./DX)
      DSH=.5*(DX-1./DX)
      IF(IJKLM.LT.0) GO TO 450
      FX=DC*DCH+ASH
      FPX=-DS*DCH+DC*DSH
      GO TO 451
  450 IF(IJKLM.EQ.-1)GO TO 452
      FX=DS/DC - DSH/DCH
      FPX=1./DC/DC -1./DCH/DCH
      GO TO 451
  452 FX= DS
      FPX=DC
  451 CONTINUE
      EP(I)=EP(I)-FX/FPX
  200 CONTINUE
C     COMPUTE MODE SHAPE CONSTANTS FROM HERE TO 1
      DO 1 N=1,NTERMS
      SN=DSIN(EP(N))
      CS=DCOS(EP(N))
      DX=DEXP(EP(N))
      SH=.5*(DX-1./DX)
      CH=.5*(DX+1./DX)
      IF(ICYL.EQ.1) GO TO 9450
      IF(IJKLM.LT.0)GO TO 460
      IF(IJKLM.GT.0) GO TO 351
C     CLAMPED - CLAMPED
C     CLAMPED - FREE
      C(1,4,3,N)=(CH*ASH+CS)/(SN*ASH+SH)*ASH
      C(1,3,3,N)=-C(1,4,3,N)
      C(1,1,3,N)= 1.
      C(1,2,3,N)= -1.
      GO TO 1
C     FREE - FREE
  351 C(1,1,3,N)= 1.
      C(1,2,3,N)= 1.
      C(1,3,3,N)= (-CS+CH)/(SN-SH)
      C(1,4,3,N)= C(1,3,3,N)
      GO TO 1
 9450 C(1,2,3,N)= DSQRT(2.D0)
      C(1,1,3,N)= 0.
      C(1,3,3,N)= 0.
      C(1,4,3,N)= 0.
      GO TO 1
  460 IF(IJKLM .EQ.-1) GO TO 453
      IF(IJKLM.EQ.-2)GO TO 454
C     CLAMPED - SIMPLE
      C(1,1,3,N)= 1.
      C(1,2,3,N)= -1.
      C(1,3,3,N)= (CS-CH)/(SH-SN)
      C(1,4,3,N)= -C(1,3,3,N)
```

```
      GO TO 1
C     SIMPLE - FREE
  454 C(1,1,3,N)= 0.
      C(1,2,3,N)= 0.
      C(1,4,3,N)= 2.*SH/(-SN+SH)
      C(1,3,3,N)= C(1,4,3,N)-2.D0
      AV= DSQRT( C(1,4,3,N) + C(1,3,3,N) )
      C(1,4,3,N)= C(1,4,3,N)/AV
      C(1,3,3,N)= C(1,3,3,N)/AV
      GO TO 1
C     SIMPLE - SIMPLE
  453 C(1,1,3,N)= 0.
      C(1,2,3,N)= 0.
      C(1,3,3,N)= 0.
      C(1,4,3,N)= DSQRT(2.D0)
    1 CONTINUE
      GO TO 701
C     ELASTIC RESTRAINT
  700 ALFA= $ALFA
      BETA= $BETA
C     FREQUENCIES AND SHAPE COEFFICIENTS ARE COMPUTED IN ELASTC.
      CALL ELASTC (CLASTC,ALFA,BETA,NTERMS)
      DO 7000 J=1,4
      DO 7000 N=1,NTERMS
 7000 C(1,J,3,N) = CLASTC(J,N)
  701 CONTINUE
C     THE COEFFICIENTS OF THE 'NORMALIZED' DERIVATIVES ARE PUT IN C(   )
      INIJ= MNIJ
      MNIJ= IDEFNE
      ID=IDEFNE
      DO 2 N=1,NTERMS
      C(2,1,3,N)= C(1,3,3,N)
      C(2,2,3,N)= C(1,4,3,N)
      C(2,3,3,N)= C(1,1,3,N)
      C(2,4,3,N)=-C(1,2,3,N)
      C(3,1,3,N)= C(1,1,3,N)
      C(3,2,3,N)=-C(1,2,3,N)
      C(3,3,3,N)= C(1,3,3,N)
    2 C(3,4,3,N)=-C(1,4,3,N)
      IF(IDEFNE.EQ.2) GO TO 9910
      DO 9900 I=1,4
      DO 9900 N=1,NTERMS
      C(1,I,1,N) = C(2,I,3,N) * EP(N)
 9900 C(1,I,2,N)=C(1,I,3,N)
      GO TO 9920
 9910 DO 9915 I=1,4
      DO 9915 N=1,NTERMS
      C(1,I,1,N)=C(1,I,3,N)
 9915 C(1,I,2,N) = C(2,I,3,N) * EP(N)
 9920 DO 9930 I=1,2
      DO 9930 N=1,NTERMS
      C(2,1,I,N)= C(1,3,I,N)
      C(2,2,I,N)= C(1,4,I,N)
      C(2,3,I,N)= C(1,1,I,N)
      C(2,4,I,N)=-C(1,2,I,N)
      C(3,1,I,N)= C(1,1,I,N)
```

```
      C(3,2,I,N)=-C(1,2,I,N)                                            SS8G224
      C(3,3,I,N)= C(1,3,I,N)                                            SS8G225
 9930 C(3,4,I,N)=-C(1,4,I,N)                                            SS8G226
C  FACTORIAL GENERATION                                                 SS8G227
      IPOWE2 = IPOWER+1                                                 SS8G228
      DO 2001 I=1,IPOWE2                                                SS8G229
      DO 2002 L=1,I                                                     SS8G230
      ALVA(I,L,2)=0.                                                    SS8G231
      J=I-1                                                             SS8G232
      K=I-L                                                             SS8G233
      DFAC = 1.                                                         SS8G234
      FAC=1.                                                            SS8G235
      IF(J.LE.1)GO TO 2003                                              SS8G236
      DO 2004 JJ=2,J                                                    SS8G237
      AMTP= JJ                                                          SS8G238
 2004 FAC= FAC*AMTP                                                     SS8G239
 2003 IF(K.LE.1)GO TO 2005                                              SS8G240
      DO 2006 KK=2,K                                                    SS8G241
      AMTP= KK                                                          SS8G242
 2006 DFAC = AMTP*DFAC                                                  SS8G243
 2005 ALVA(I,L,1)= ((-1.)**(L+1))*FAC/DFAC                              SS8G244
 2002 CONTINUE                                                          SS8G245
 2001 ALVA(I,I,2)=ALVA(I,I,1)                                           SS8G246
      PI=3.1415926535898/2.                                             SS8G247
      DO 1001 IUVW=1,3                                                  SS8G248
      DO 1001 JUVW=1,3                                                  SS8G249
      DO 1001 M=1,NTERMS                                                SS8G250
      EPM = EP(M)                                                       SS8G251
      DO 1001 N=1,NTERMS                                                SS8G252
      EPN= EP(N)                                                        SS8G253
      OMEGA1= EPM+ EPN                                                  SS8G254
      OMEGA2= EPN- EPM                                                  SS8G255
      EX1= .25*DEXP(OMEGA1)                                             SS8G256
      EMX1=1./EX1/16.                                                   SS8G257
      EX2 =.25*DEXP(OMEGA2)                                             SS8G258
      EMX2=1./EX2/16.                                                   SS8G259
      SN1 = DSIN(OMEGA1)/2.                                             SS8G260
      SN2 = DSIN(OMEGA2)/2.                                             SS8G261
      CS1=DCOS(OMEGA1)/2.                                               SS8G262
      CS2=DCOS(OMEGA2)/2.                                               SS8G263
      FACTOR=1.                                                         SS8G264
      DO 1002 I=1,IPOWER                                                SS8G265
      FACTOR= FACTOR*I                                                  SS8G266
      O1I = (OMEGA1)**I                                                 SS8G267
      FFF(I) = EX1/O1I                                                  SS8G268
      T111= 0.0                                                         SS8G269
      T112= ((-1.)**I)*EMX1/O1I                                         SS8G270
      T113= (1.-(-1.)**(I+1))/2./O1I /2.                                SS8G271
 1003 T121=0.                                                           SS8G272
      T122=(DSIN(I*PI)*SN1 +DCOS(I*PI)*CS1)/O1I                         SS8G273
      T123= DCOS(I*PI)/2./O1I                                           SS8G274
      IF(M.EQ.N) GO TO 1004                                             SS8G275
      O2I= (OMEGA2)**I                                                  SS8G276
      T211= EX2/O2I                                                     SS8G277
      T212= ((-1.)**I)*EMX2/O2I                                         SS8G278
      T213= (1.-(-1.)**(I+1))/4./O2I                                    SS8G279
```

```
      IF(DABS(T211).GE. DABS(T212)) GO TO 1005
      TX15= T211
      T211= T212
      T212= TX15
 1005 T221= 0.
      T222= (DSIN(I*PI)*SN2 + DCOS(I*PI)*CS2)/O2I
      T223= DCOS(I*PI)/2./O2I
      GO TO 1006
 1004 T211= 0.
      T221= 0.
      T212= .5/FACTOR
      T222= T212
      T213=0.
      T223= 0.
 1006 TH(I,1,1,1) = T111 + T211
      TH(I,1,1,2) = T112 + T212
      TH(I,2,2,1) = T121 + T221
      TH(I,2,2,2) = T122 + T222
      TH(I,3,3,1) = T111 - T211
      TH(I,3,3,2) = T112 - T212
      TH(I,4,4,1) =-T121 + T221
      TH(I,4,4,2) =-T122 + T222
      TH(I,1,1,3) = T113 + T213
      TH(I,2,2,3) = T123 + T223
      TH(I,3,3,3) = T113 - T213
 1002 TH(I,4,4,3) =-T123 + T223
      IFLAG= -1
 1007 EPSAVE = EPN
      EPN = EPM
      EPM = EPSAVE
      OMEGA1 =EPM+EPN
      OMEGA2 =EPN-EPM
      EX1= .25*DEXP(OMEGA1)
      EMX1= 1./EX1/16.
      EX2 = DEXP(OMEGA2)/4.
      EMX2= 1./EX2/16.
      SN1= DSIN(OMEGA1)/2.
      SN2= DSIN(OMEGA2)/2.
      CS1= DCOS(OMEGA1)/2.
      CS2= DCOS(OMEGA2)/2.
      DELO= EPM*EPM+ EPN*EPN
      DELI1= 1.
      DELI2= 0.
      EPEPN =  DEXP(EPN)/2.
      EMEPN =  1./EPEPN/4.
      SNEPM= DSIN(EPM)
      CSEPM= DCOS(EPM)
      DO 1008 I=1,IPOWER
      DELI1S = EPN*DELI1 - EPM*DELI2
      DELI2 = EPM*DELI1 + EPN*DELI2
      DELI1 = DELI1S
      O1I = (OMEGA1)**I
      DELOI =(DELO)**I
      TH(I,3,1,1) = 0.0
      TH(I,3,1,2)=((-1.)**(I+1))*EMX1/O1I
      TH(I,3,1,3)=  (1.-(-1.)**I)/2./O1I/2.
```

```
      TH(I,4,2,1) =0.
      TH(I,4,2,2) = (-DSIN(I*PI)*CS1 +DCOS(I*PI)*SN1)/O1I
      TH(I,4,2,3) = -DSIN(I*PI)/2./O1I
      TH(I,1,2,1) = EPEPN/DELOI*(DELI1*CSEPM + DELI2*SNEPM)
      TH(I,1,2,2) = EMEPN/DELOI*(((-1.)**I)*DELI1*CSEPM
     1 +((-1.)**(I+1))*DELI2*SNEPM)
      TH(I,1,2,3) = DELI1/2./DELOI*(1.+(-1.)**I)
      TH(I,3,2,3) = DELI1/2./DELOI*(1.-(-1.)**I)
      TH(I,3,2,1) = TH(I,1,2,1)
      TH(I,3,2,2) =-TH(I,1,2,2)
      TH(I,1,4,1) = EPEPN/DELOI*(DELI1*SNEPM -DELI2*CSEPM)
      TH(I,1,4,2) = EMEPN/DELOI*(((-1.)**I)*DELI2*CSEPM
     1 +((-1.)**I)*DELI1*SNEPM)
      TH(I,1,4,3) = DELI2/2./DELOI*(-1.+(-1.)**I)
      TH(I,3,4,3) = DELI2/2./DELOI*(-1.-(-1.)**I)
      TH(I,3,4,1)= TH(I,1,4,1)
      TH(I,3,4,2)=-TH(I,1,4,2)
      IF(M.EQ.N) GO TO 1009
      O2I=(OMEGA2)**I
      TBIG =EX2/O2I
      TSMALL= ((-1.)**(I+1))*EMX2/O2I
      TH(I,3,1,3) = TH(I,3,1,3) + (1.-(-1.)**I)/2./O2I/2.
      TH(I,4,2,2) = TH(I,4,2,2)+ (-DSIN(I*PI)*CS2+DCOS(I*PI)*SN2)/O2I
      TH(I,4,2,3) = TH(I,4,2,3) -DSIN(I*PI)/2./O2I
      IF(DABS(TBIG).GE.DABS(TSMALL))GO TO 1010
      TX15 =TBIG
      TBIG = TSMALL
      TSMALL = TX15
 1010 TH(I,3,1,1) = TH(I,3,1,1) + TBIG
      TH(I,3,1,2) = TH(I,3,1,2) + TSMALL
 1009 CONTINUE
 1008 CONTINUE
      IF(IFLAG.GT. 0) GO TO 1011
      IFLAG = +1
      DO 1012 I=1,IPOWER
      DO 1012 J=1,3
      TH(I,1,3,J) = TH(I,3,1,J)
      TH(I,2,4,J)= TH(I,4,2,J)
      TH(I,2,1,J)= TH(I,1,2,J)
      TH(I,2,3,J)= TH(I,3,2,J)
      TH(I,4,1,J)= TH(I,1,4,J)
 1012 TH(I,4,3,J)= TH(I,3,4,J)
      GO TO 1007
 1011 CONTINUE
C     TH(I,K,J) ARE NOW STORED
      DO 1001 K=1,6
      IF(K-2)25,26,27
   27 IF(K-4)28,29,30
   30 IF(K-6)31,32,32
   25 NN=1
      MM=1
      GO TO 6
   26 NN=2
      MM=2
      GO TO 6
   28 NN=3
```

```
      MM=3
      GO TO 6
   29 NN=2
      MM=1
      GO TO 6
   31 NN=3
      MM=1
      GO TO 6
   32 NN=3
      MM=2
    6 DO 7 J=1,4
      CN(J)=C(NN,J,IUVW,N)
    7 CM(J)=C(MM,J,JUVW,M)
      EXYZ = (EPN**(NN-1))*(EPM**(MM-1))
      DO 8 J=1,4
      DO 8 I=1,4
    8 E(J,I)= CN(J)*CM(I)*EXYZ
      SAVEIT= (CN(1)+CN(3))*(CM(1)+CM(3))*EXYZ
      I = 1
      AL(I,K,IUVW,N,JUVW,M)=0.
      SAVE1= 0.
      SAVE2= 0.
      SAVE3= 0.
      SAVE4=0.
      DO 1114 L=1,I
      SAVE1 = SAVE1 + SAVEIT*ALVA(I,L,1)*FFF(L)
      DO 1114 IT=1,4
      DO 1114 IU=1,4
      SAVE4= SAVE4 +E(IT,IU)*ALVA(I,L,1)*TH(L,IT,IU,1)
      SAVE2= SAVE2 +E(IT,IU)*ALVA(I,L,1)*TH(L,IT,IU,2)
 1114 SAVE3= SAVE3 +E(IT,IU)*ALVA(I,L,2)*TH(L,IT,IU,3)
 1014 AL(I,K,IUVW,N,JUVW,M)= SAVE1 + SAVE2 - SAVE3 + SAVE4
      IF ( K .LE. 2 ) KK=K
      IF ( K .EQ. 3 ) GO TO 1001
      IF ( K .EQ. 4 ) KK=3
      IF ( K .GE. 5 ) GO TO 1001
      IF(IUVW.NE.3) GO TO 1001
      IF(JUVW.NE.3) GO TO 1001
      DO 6000 I=1,IPOWER
      $W(I,ID,KK,N,M) = 0.
      SAVE1 = 0.
      SAVE2 = 0.
      SAVE3 = 0.
      SAVE4 = 0.
      DO 5000 L=1,I
      SAVE1 = SAVE1 + SAVEIT * ALVA(I,L,1) * FFF(L)
      DO 5000 IT=1,4
      DO 5000 IU=1,4
      SAVE4 = SAVE4 + E(IT,IU) * ALVA(I,L,1) * TH(L,IT,IU,1)
      SAVE2= SAVE2 + E(IT,IU) * ALVA(I,L,1) * TH(L,IT,IU,2)
 5000 SAVE3 = SAVE3 + E(IT,IU) * ALVA(I,L,2) * TH(L,IT,IU,3)
 6000 $W(I,ID,KK,N,M) = SAVE1 + SAVE2 - SAVE3 + SAVE4
 1001 CONTINUE
C     THE P INTEGRALS ARE NOW EVALUATED, AND ALSO ANY SPECIAL CASES.
      IPO2= IPOWER+1
      IN = 1
```

```
      IF ( INIJ .EQ. 5 .AND. IZ .EQ. 1 )  IN = INIJ - 3
      IF ( INIJ .EQ. 6 .AND. IZ .EQ. 0 )  IN = INIJ - 3
      DO 811 NUVW=1,3
      DO 806 I=1,4
      DO 806 J=1,NTERMS
  806 CE(I,J)=C(1,I,NUVW,J)
      CALL PPP (IN,NTERMS,IPOWER,ID,NUVW, 1 )
      DO 807 I=1,IPO2
      DO 807 J=1,NTERMS
      IF ( IN .EQ. 1 )  GO TO 807
      PZ(I,1,NUVW,J) = P(I,J)
  807 $P(I,ID,1,NUVW,J) = P(I,J)
      DO 808 I=1,4
      DO 808 J=1,NTERMS
  808 CE(I,J)=C(2,I,NUVW,J)*EP(J)
      CALL PPP (IN,NTERMS,IPOWER,ID,NUVW, 2 )
      DO 809 I=1,IPO2
      DO 809 J=1,NTERMS
      IF ( IN .EQ. 1 )  GO TO 809
      PZ(I,2,NUVW,J) = P(I,J)
  809 $P(I,ID,2,NUVW,J) = P(I,J)
      DO 810 I=1,4
      DO 810 J=1,NTERMS
  810 CE(I,J)=C(3,I,NUVW,J)*EP(J)*EP(J)
      CALL PPP (IN,NTERMS,IPOWER,ID,NUVW, 3 )
      DO 811 I=1,IPO2
      DO 811 J=1,NTERMS
      IF ( IN .EQ. 1 )  GO TO 811
      PZ(I,3,NUVW,J) = P(I,J)
  811 $P(I,ID,3,NUVW,J) = P(I,J)
      IF ( IN .EQ. 1 )  GO TO 805
      CALL SPECAL (IPOWER,NTERMS,INIJ,IDEFNE)
      IN=IN-1
  805 I=1
      DO 33 IUVW=1,3
      DO 33 JUVW=1,3
      DO 33 K=1,6
      DO 33 N=1,NTERMS
      DO 33 M=1,NTERMS
      $AL(I,MNIJ,K,IUVW,N,JUVW,M)=AL(I,K,IUVW,N,JUVW,M)
   33 CONTINUE
C     THE MODE SHAPES AND ITS DERIVATIVES ARE EVALUATED AT 25 PCINTS.
      DO 707 N=1,3
      DO 40 J=1   ,NTERMS
      C(4,1,N,J)=C(3,3,N,J)
      C(4,2,N,J)=C(3,4,N,J)
      C(4,3,N,J)=C(3,1,N,J)
      C(4,4,N,J)=-C(3,2,N,J)
      DO 2750 I=1,4
      SAVE1 = C(I,1,N,J)
      C(I,1,N,J)=C(I,1,N,J)+C(I,3,N,J)
 2750 C(I,3,N,J)=C(I,3,N,J) - SAVE1
      DO 40 L=1,25
      YU=L-1
      YU=YU/24.
      AA=DEXP(EP(J)*YU)
```

```
      CN(1)=.5*(AA       )                                              SS8G504
      CN(3)=.5*(   -1./AA)                                              SS8G505
      CN(2)=DCOS(EP(J)*YU)                                              SS8G506
      CN(4)=DSIN(EP(J)*YU)                                              SS8G507
      DO 40 I=1,4                                                       SS8G508
      EVAL(I,N,J,L)=0.D0                                                SS8G509
      DO 40 K=1,4                                                       SS8G510
   40 EVAL(I,N,J,L)=EVAL(I,N,J,L)+CN(K)*C(I,K,N,J)*(EP(J)**(I-1))       SS8G511
      IF ( INIJ .EQ. 5 .AND. IZ .EQ. 1 )  GO TO 816                     SS8G512
      IF ( INIJ .EQ. 6 .AND. IZ .EQ. 0 )  GO TO 816                     SS8G513
      GO TO 815                                                         SS8G514
  816 DO 817 J=1,IN                                                     SS8G515
      DO 817 L=1,25                                                     SS8G516
      DO 817 I=1,4                                                      SS8G517
  817 EVAL(I,N,J,L)=EVQ(I,N,J,L)                                        SS8G518
  815 CONTINUE                                                          SS8G519
   41 CONTINUE                                                          SS8G520
      DO 707 K=1,NTERMS                                                 SS8G521
      DO 707 L=1,25                                                     SS8G522
      DO 707 I=1,4                                                      SS8G523
  707 $EVAL(I,MNIJ,N,K,L)=EVAL(I,N,K,L)                                 SS8G524
      IF ( MNIJ .EQ. 1 )  GO TO 59                                      SS8G525
      IF ( NSTRNG .EQ. 0 ) GO TO 90                                     SS8G526
      DO 50 L=1,NSTRNG                                                  SS8G527
      DO 50 J=1,NTERMS                                                  SS8G528
      Y = $STRNG(L)                                                     SS8G529
      AA = DEXP(EP(J)*Y)                                                SS8G530
      CN(1) = .5*AA                                                     SS8G531
      CN(3) = -.5/AA                                                    SS8G532
      CN(2) = DCOS(EP(J)*Y)                                             SS8G533
      CN(4) = DSIN(EP(J)*Y)                                             SS8G534
      $ESV(J,L) = 0.                                                    SS8G535
      $ESW(J,L) = 0.                                                    SS8G536
      $ESDW(J,L)= 0.                                                    SS8G537
      DO 50 K=1,4                                                       SS8G538
      $ESV(J,L) = $ESV(J,L) + CN(K) * C(1,K,2,J)                        SS8G539
      $ESW(J,L) = $ESW(J,L) + CN(K) * C(1,K,3,J)                        SS8G540
   50 $ESDW(J,L)= $ESDW(J,L)+ CN(K) * C(2,K,3,J)*EP(J)                  SS8G541
      IF ( INIJ .NE. 5 ) GO TO 52                                       SS8G542
      IF ( IZ .NE. 1 )  GO TO 52                                        SS8G543
      DO 51 L=1,NSTRNG                                                  SS8G544
      $ESV(1,L) = S3                                                    SS8G545
      $ESW(1,L) = S3 * $STRNG(L)                                        SS8G546
   51 $ESDW(1,L) = S3                                                   SS8G547
      GO TO 90                                                          SS8G548
   52 IF ( INIJ .NE. 6 )  GO TO 90                                      SS8G549
      IF ( IZ .NE. 0 )  GO TO 90                                        SS8G550
      DO 53 L=1,NSTRNG                                                  SS8G551
      $ESV(1,L) = 0.                                                    SS8G552
      $ESV(2,L) = -2.*S3                                                SS8G553
      $ESW(1,L) = 1.                                                    SS8G554
      $ESW(2,L) = S3 * ( 1. - 2. * $STRNG(L) )                          SS8G555
      $ESDW(1,L)= 0.                                                    SS8G556
   53 $ESDW(2,L)= -2.*S3                                                SS8G557
      GO TO 90                                                          SS8G558
   59 IF ( NRING .EQ. 0 )  GO TO 90                                     SS8G559
```

```
      DO 60 L=1,NRING
      DO 60 J=1,NTERMS
      X = $RINGS(L)
      AA = DEXP(EP(J)*X)
      CN(1) = .5*AA
      CN(3) = -.5/AA
      CN(2) = DCOS(EP(J)*X)
      CN(4) = DSIN(EP(J)*X)
      $ERU(J,L) = 0.
      $ERW(J,L) = 0.
      $ERDW(J,L) = 0.
      DO 60 K=1,4
      $ERU(J,L) = $ERU(J,L) + CN(K) * C(1,K,1,J)
      $ERW(J,L) = $ERW(J,L) + CN(K) * C(1,K,3,J)
   60 $ERDW(J,L)= $ERDW(J,L)+ CN(K) * C(2,K,3,J)*EP(J)
      IF ( INIJ .NE. 5 )  GO TO 62
      IF ( IZ .NE. 1 )  GO TO 62
      DO 61 L=1,NRING
      $ERU(1,L) = S3
      $ERW(1,L) = S3*$RINGS(L)
   61 $ERDW(1,L)= S3
      GO TO 90
   62 IF ( INIJ .NE. 6 ) GO TO 90
      IF ( IZ .NE. 0 )  GO TO 90
      DO 63 L=1,NRING
      $ERU(1,L) = 0.
      $ERU(2,L) = -2.*S3
      $ERW(1,L) = 1.
      $ERW(2,L) = S3*(1.-2.*$RINGS(L))
      $ERDW(1,L)= 0.
   63 $ERDW(2,L)= -2.*S3
   90 CONTINUE
      RETURN
      END
```

CC = 00594

```
      SUBROUTINE ELASTC(RETURN,ALFA,BETA,N)
C     THIS SUBROUTINE COMPUTES THE FREQUENCIES (STORED IN EP( ) ) AND
C     MODE SHAPES FOR A BEAM WITH ELASTIC MOMENT RESTRAINT AT BOTH ENDS.
C     THE MODE SHAPES ARE DEFINED BY MEANS OF FOUR CONSTANTS FOR EACH
C     FREQUENCY, WHICH ARE RETURNED IN THE ARRAY NAMED RETURN( ).  THE
C     RESTRAINT IS SPECIFIED IN TERMS OF THE INPUT QUANTITIES ALPHA AND
C     BETA.  AT THE ZERO END, THE RESTRAINED BOUNDARY CONDITION IS THAT
C     THE SLOPE = ALFA*CURVATURE.  AT THE OTHER END, THE CONSTANT OF
C     PROPORTIONALITY IS -BETA.  THE ROOTS OF THE CHARACTERISTIC
C     EQUATION ARE FOUND IN DOUBLE PRECISION USING AN INTERVAL HALFING
C     TECHNIQUE.  THE INTERVAL IS HALVED 70 TIMES, SO THAT THE FINAL
C     INTERVAL IS 1.6/(2**70)
C **  REVISED FOR CURVED PANELS  - -  8/69
      IMPLICIT REAL*8(A-H,O-Z), INTEGER(I-N)
      COMMON / BLOCK /        AL(1,6,3,10,3,10),  EVAL(4,3,10,25),
     1                 EVQ(4,3,2,25),      PZ(11,3,3,10),
     2                 TH(10,4,4,3),       ALVA(11,11,2),      P(11,10),
     3                 CE(4,10), E(4,4),   EP(10),   CN(4),    CM(4)
      DIMENSION C(2,4), F(4), RETURN(4,10)
      BETA = -BETA
      AA=1.
      C(1,3)=0.
      C(1,4)=1.
      C(2,3)=1.
      C(2,4)=0.
      DO 4 L=1,N
      ELEFT=L
      ELEFT=ELEFT*3.1415
      ERIGHT=ELEFT+1.6
      I=0
      EPZ=ELEFT
      GO TO 13
   11 ELEFX=PTE
   12 EPZ=(ELEFT+ ERIGHT)/2.
   13 I=I+1
      G1=ALFA/2./EPZ
      G4=G1*BETA
      C(1,1)=G1
      C(1,2)= -G1
      C(2,1)=G1
      C(2,2)=-G1
      EX=DEXP(EPZ)
      EXX=1./EX
      F(1)=.5*(EX+EXX)
      F(2)=DCOS(EPZ)
      F(3)=.5*(EX-EXX)
      F(4)=DSIN(EPZ)
      PTE=  -G4*2.*(1. -F(1)*F(2)) +(ALFA -BETA) *( F(1)*F(4) - F(3)*
     1F(2)) + 2.*EPZ*F(3)*F(4)
      IF(I.LT.2) GO TO 11
      IF(PTE*ELEFX)16,17,18
   16 ERIGHT = EPZ
      GO TO 19
   18 ELEFT= EPZ
      ELEFX = PTE
   19 IF(I.LT.30)GO TO 12
```

```
17 CONTINUE
 8 PTE =0.
   PJA= 0.
   DO 9 J=1,4
   PTE= PTE+ C(2,J)*F(J)
 9 PJA= PJA+ C(1,J)*F(J)
   CC=-PJA/PTE
   BB=-(AA+CC)*G1
   DD=-BB
   RETURN(1,L) = DD
   RETURN(2,L) = BB
   RETURN(3,L) = CC
   RETURN(4,L) = AA
   EP(L) = EPZ
 4 CONTINUE
   RETURN
   END
```

CC = 00073

```
      SUBROUTINE PPP (IN,NTERMS,IPOWER,ID,NUVW, IR )
C     THIS SUBROUTINE COMPUTES THE 'P' INTEGRALS--THE INTEGRALS OF A
C     SINGLE MODE SHAPE OR ITS DERIVATIVE.  THE INPUT IS IN (THE NUMBER
C     OF SPECIAL CASES PLUS ONE), THE ARRAY CE( ) WHICH CONTAINS THE
C     FOUR COEFFICIENTS OF THE MODE SHAPE (OR ITS DERIVATIVE) WHICH IS
C     TO BE INTEGRATED. THE OUTPUT IS THE ARRAY P( ) CONTAINING THE
C     INTEGRALS. THE ROUTINE ALSO NEEDS THE VALUES OF EP( ), ENTERED
C     THROUGH COMMON. THE ROUTINE IS IN DOUBLE PRECISION.
C **  REVISED FOR CURVED PANELS  - -  8/69
      IMPLICIT REAL*8(A-H,O-Z),INTEGER  (I-N)
      COMMON / BLOCK /        AL(1,6,3,10,3,10),  EVAL(4,3,10,25),
     1              EVQ(4,3,2,25),       PZ(11,3,3,10),
     2              TH(10,4,4,3),        ALVA(11,11,2),       P(11,10),
     3              CE(4,10), E(4,4),    EP(10),    CN(4),    CM(4)
      DIMENSION G(4), C(12,4,10), F(4)
      IPOW2=IPOWER+1
      PETE=3.
      AQB= -1.
      S3=DSQRT(PETE)
      IF(IN.EQ.3) GO TO 60
      IF(IN.EQ.2) GO TO 50
      IF(IN.EQ.1) GO TO 61
C     SPECIAL CASES ARE COMPUTED FIRST.
   50 IF(NUVW.EQ.3) GO TO 210
      IF(ID.NE.1)    GO TO 230
      IF(NUVW.NE.1) GO TO 210
  190 DO 200 I=1,IPOW2
      T=I
      P(I,1) = S3/T
      IF ( IR .NE. 1 ) P(I,1) = 0.D0
  200 CONTINUE
      GO  TO 61
  210 DO 220 I=1,IPOW2
      T=I+1
      IF ( IR .EQ. 1 ) P(I,1) = S3/T
      IF ( IR.EQ. 2 ) P(I,1) = S3/(T-1.)
      IF ( IR .EQ. 3 ) P(I,1) = 0.D0
  220 CONTINUE
      GO TO 61
  230 IF(NUVW.EQ.1) GO TO 210
      GO TO 190
   60 IF (NUVW.EQ.3) GO TO 310
      IF (ID.NE.1)    GO TO 330
      IF (NUVW.NE.1) GO TO 310
  290 DO 300 I=1,IPOW2
      T=I
      P(I,1) = 0.D0
      P(I,2) = 0.D0
      IF ( IR .EQ. 1 ) P(I,2) = -2.D0*S3/T
  300 CONTINUE
      GO TO 61
  310 DO 320 I=1,IPOW2
      T=I
      TT= 1./T -2./(T+1.)
      P(I,1) = 0.D0
      P(I,2) = 0.D0
```

```
      IF ( IR .EQ. 1 )  P(I,1) = 1.D0/T
      IF ( IR .EQ. 1 )  P(I,2) = S3*TT
      IF ( IR .EQ. 2 )  P(I,2) = -2.D0*S3/T
  320 CONTINUE
      GO TO 61
  330 IF (NUVW.EQ.1)  GO TO 310
      GO TO 290
   61 INN=  IN
      G(1)=1.
      G(2)=1.
      G(3)=0.
      G(4)=0.
      DO 1 L=INN,NTERMS
      EX=DEXP(EP(L))
      SH=.5*(EX-1./EX)
      SN=DSIN(EP(L))
      CS=DCOS(EP(L))
      CH=.5*(EX+1./EX)
      DO 2 J=1,IPOW2,2
      C(J,1,L)=CE(3,L)/(EP(L)**J)
      C(J,3,L)=CE(1,L)/(EP(L)**J)
      C(J+1,1,L)=CE(1,L)/(EP(L)**(J+1))
    2 C(J+1,3,L)=CE(3,L)/(EP(L)**(J+1))
      IJK=0
      DO 10 J=1,IPOW2,2
      IJK= IJK+ 1
      C(J,2,L)= (AQB**IJK)*CE(4,L)/(EP(L)**J)
      C(J+1,2,L)= (AQB**IJK)*CE(2,L)/(EP(L)**(J+1))
      C(J,4,L )=-(AQB** IJK    )*CE(2,L)/(EP(L)**J)
   10 C(J+1,4,L)= (AQB**(IJK))*CE(4,L)/(EP(L)**(J+1))
      F(1)=CH
      F(2)=CS
      F(3)=SH
      F(4)=SN
      DO 4 I=1,IPOW2
    4 P(I,L)=0.
      DO 1 I=1,4
      DO 1 JJ=1,IPOW2
      DO 100 KK =1,JJ
  100 P(JJ,L) = P(JJ,L) + C(KK,I,L)*F(I)*ALVA(JJ,KK,1)
    1 P(JJ,L) = P(JJ,L) - C(JJ,I,L)*G(I)*ALVA(JJ,JJ,2)
      RETURN
      END
```

CC = 00099

```
      SUBROUTINE SPECAL (IPOWER,NTERMS,MNIJ,IDEFNE)
C     THIS SUBROUTINE COMPUTES THE 'SPECIAL' CASES INTEGRALS FOR FREE-
C     FREE AND SIMPLE FREE BEAM SHAPES.  THE INPUT NECESSARY IS THE
C     'P' INTEGRALS FROM SUBROUTINE PPP FOR THE CONDITION THE SUBROUTINE
C     IS BEING USED FOR (MNIJ=5 FOR SIMPLE-FREE, 6 FOR FREE-FREE).  THE
C     SUBROUTINE RETURNS THE INTEGRALS IN THE ARRAY ALL.  THE MODE SHAPE
C     EVALUATIONS, AND DERIVATIVE EVALUATIONS, FOR THE SPECIAL CASES ARE
C     MADE AND RETURNED IN EVQ.  THE ROUTINE IS IN DOUBLE PRECISION.
C **  REWRITTEN FOR CURVED PANELS  - -  8/69
      IMPLICIT REAL*8(A-H,O-Z), INTEGER  (I-N)
      COMMON / BLOCK /        AL(  6,3,10,3,10),  EXAL(4,3,10,25),
     1                EVAL(4,3,2,25),      P(11,3,3,10),
     2                TH(10,4,4,3),        ALVA(11,11,2),       PDUM(11,10),
     3                CE(4,10), E(4,4),    EP(10),    CN(4),     CM(4)
      COMMON / ARRAYS / $P(11,2,3,3,10),  $AL(1,2,6,3,10,3,10),
     1                      $W(10,2,3,10,10)
      ID=IDEFNE
      IF(ID.EQ.1) JD=2
      IF(ID.EQ.2) JD=1
      S3 = DSQRT ( 3.D0 )
C  THE INTEGRALS ARE EVALUATED FROM HERE TO STATEMENT 1 .
      I = 1
      T=I
C  SIMPLE - FREE  BOUNDARY CONDITION
      IF (MNIJ.NE.5) GO TO 200
      DO 90 K=1,6
      DO 90 IUVW=1,3
      DO 90 JUVW=1,3
      DO 90 M=1,NTERMS
      AL(K,IUVW,1,JUVW,M) = 0.D0
   90 AL(K,IUVW,M,JUVW,1) = 0.D0
      AL(1,ID,1,ID,1)= 3.D0
      AL(1,ID,1,ID,1)= 1.5D0
      AL(1,ID,1,3 ,1)= 1.5D0
      AL(1,JD,1,ID,1)= 1.5D0
      AL(4,JD,1,ID,1)= 3.D0
      AL(1,JD,1,JD,1)= 1.D0
      AL(2,JD,1,JD,1)= 3.D0
      AL(4,JD,1,JD,1)= 1.5D0
      AL(1,JD,1,3 ,1)= 1.D0
      AL(2,JD,1,3 ,1)= 3.D0
      AL(4,JD,1,3 ,1)= 1.5D0
      AL(1,3 ,1,ID,1)= 1.5D0
      AL(4,3 ,1,ID,1)= 3.D0
      AL(1,3 ,1,JD,1)= 1.D0
      AL(2,3 ,1,JD,1)= 3.D0
      AL(4,3 ,1,JD,1)= 1.5D0
      AL(1,3 ,1,3 ,1)= 1.D0
      AL(2,3 ,1,3 ,1)= 3.D0
      AL(4,3 ,1,3 ,1)= 1.5D0
      IF ( NTERMS.EQ.1 )  GO TO 101
      DO 100 M=2,NTERMS
      AL(1,ID,1,ID,M)= S3*P(1,1,ID,M)
      AL(1,ID,1,JD,M)= S3*P(1,1,JD,M)
      AL(1,ID,1,3 ,M)= S3*P(1,1,3 ,M)
      AL(1,ID,M,ID,1)= S3*P(1,1,ID,M)
```

```
      AL(4,ID,M,ID,1)= S3*P(1,2,ID,M)                                   SS8J056
      AL(5,ID,M,ID,1)= S3*P(1,3,ID,M)                                   SS8J057
      AL(1,ID,M,JD,1)= S3*P(2,1,ID,M)                                   SS8J058
      AL(2,ID,M,JD,1)= S3*P(1,2,ID,M)                                   SS8J059
      AL(4,ID,M,JD,1)= S3*P(2,2,ID,M)                                   SS8J060
      AL(5,ID,M,JD,1)= S3*P(2,3,ID,M)                                   SS8J061
      AL(6,ID,M,JD,1)= S3*P(1,3,ID,M)                                   SS8J062
      AL(1,ID,M,3 ,1)= S3*P(2,1,ID,M)                                   SS8J063
      AL(2,ID,M,3 ,1)= S3*P(1,2,ID,M)                                   SS8J064
      AL(4,ID,M,3 ,1)= S3*P(2,2,ID,M)                                   SS8J065
      AL(5,ID,M,3 ,1)= S3*P(2,3,ID,M)                                   SS8J066
      AL(6,ID,M,3 ,1)= S3*P(1,3,ID,M)                                   SS8J067
      AL(1,JD,1,ID,M)= S3*P(2,1,ID,M)                                   SS8J068
      AL(2,JD,1,ID,M)= S3*P(1,2,ID,M)                                   SS8J069
      AL(4,JD,1,ID,M)= S3*P(1,1,ID,M)                                   SS8J070
      AL(1,JD,1,JD,M)= S3*P(2,1,JD,M)                                   SS8J071
      AL(2,JD,1,JD,M)= S3*P(1,2,JD,M)                                   SS8J072
      AL(4,JD,1,JD,M)= S3*P(1,1,JD,M)                                   SS8J073
      AL(1,JD,1,3 ,M)= S3*P(2,1,3 ,M)                                   SS8J074
      AL(2,JD,1,3 ,M)= S3*P(1,2,3 ,M)                                   SS8J075
      AL(4,JD,1,3 ,M)= S3*P(1,1,3 ,M)                                   SS8J076
      AL(1,JD,M,ID,1)= S3*P(1,1,JD,M)                                   SS8J077
      AL(4,JD,M,ID,1)= S3*P(1,2,JD,M)                                   SS8J078
      AL(5,JD,M,ID,1)= S3*P(1,3,JD,M)                                   SS8J079
      AL(1,JD,M,JD,1)= S3*P(2,1,JD,M)                                   SS8J080
      AL(2,JD,M,JD,1)= S3*P(1,2,JD,M)                                   SS8J081
      AL(4,JD,M,JD,1)= S3*P(2,2,JD,M)                                   SS8J082
      AL(5,JD,M,JD,1)= S3*P(2,3,JD,M)                                   SS8J083
      AL(6,JD,M,JD,1)= S3*P(1,3,JD,M)                                   SS8J084
      AL(1,JD,M,3 ,1)= S3*P(2,1,JD,M)                                   SS8J085
      AL(2,JD,M,3 ,1)= S3*P(1,2,JD,M)                                   SS8J086
      AL(4,JD,M,3 ,1)= S3*P(2,2,JD,M)                                   SS8J087
      AL(5,JD,M,3 ,1)= S3*P(2,3,JD,M)                                   SS8J088
      AL(6,JD,M,3 ,1)= S3*P(1,3,JD,M)                                   SS8J089
      AL(1,3 ,1,ID,M)= S3*P(2,1,ID,M)                                   SS8J090
      AL(2,3 ,1,ID,M)= S3*P(1,2,ID,M)                                   SS8J091
      AL(4,3 ,1,ID,M)= S3*P(1,1,ID,M)                                   SS8J092
      AL(1,3 ,1,JD,M)= S3*P(2,1,JD,M)                                   SS8J093
      AL(2,3 ,1,JD,M)= S3*P(1,2,JD,M)                                   SS8J094
      AL(4,3 ,1,JD,M)= S3*P(1,1,JD,M)                                   SS8J095
      AL(1,3 ,1,3 ,M)= S3*P(2,1,3 ,M)                                   SS8J096
      AL(2,3 ,1,3 ,M)= S3*P(1,2,3 ,M)                                   SS8J097
      AL(4,3 ,1,3 ,M)= S3*P(1,1,3 ,M)                                   SS8J098
      AL(1,3 ,M,ID,1)= S3*P(1,1,3 ,M)                                   SS8J099
      AL(4,3 ,M,ID,1)= S3*P(1,2,3 ,M)                                   SS8J100
      AL(5,3 ,M,ID,1)= S3*P(1,3,3 ,M)                                   SS8J101
      AL(1,3 ,M,JD,1)= S3*P(2,1,3 ,M)                                   SS8J102
      AL(2,3 ,M,JD,1)= S3*P(1,2,3 ,M)                                   SS8J103
      AL(4,3 ,M,JD,1)= S3*P(2,2,3 ,M)                                   SS8J104
      AL(5,3 ,M,JD,1)= S3*P(2,3,3 ,M)                                   SS8J105
      AL(6,3 ,M,JD,1)= S3*P(1,3,3 ,M)                                   SS8J106
      AL(1,3 ,M,3 ,1)= S3*P(2,1,3 ,M)                                   SS8J107
      AL(2,3 ,M,3 ,1)= S3*P(1,2,3 ,M)                                   SS8J108
      AL(4,3 ,M,3 ,1)= S3*P(2,2,3 ,M)                                   SS8J109
      AL(5,3 ,M,3 ,1)= S3*P(2,3,3 ,M)                                   SS8J110
      AL(6,3 ,M,3 ,1)= S3*P(1,3,3 ,M)                                   SS8J111
```

```
  100 CONTINUE
  101 CONTINUE
      DO 122 I=1,IPOWER
      T = I
      $W(I,ID,1,1,1) = 3./(2.+T)
      $W(I,ID,2,1,1) = 3./T
      $W(I,ID,3,1,1) = 3./(1.+T)
      IF ( NTERMS.EQ.1 )  GO TO 125
      DO 122 M=2,NTERMS
      Z = S3*P(I+1,1,3,M)
      $W(I,ID,1,1,M) = Z
      $W(I,ID,1,M,1) = Z
      Z = S3*P(I,2,3,M)
      $W(I,ID,2,1,M) = Z
      $W(I,ID,2,M,1) = Z
      $W(I,ID,3,1,M) = S3*P(I,1,3,M)
      $W(I,ID,3,M,1) = S3*P(I+1,2,3,M)
  122 CONTINUE
      GO TO 125
C   FREE - FREE  BOUNDARY CONDITION
  200 CONTINUE
      DO 205 K=1,6
      DO 205 KUVW=1,2
      DO 205 IUVW=1,3
      DO 205 JUVW=1,3
      DO 205 M=1,NTERMS
      AL(   K,IUVW,KUVW,JUVW,M)=0.D0
      AL(   K,IUVW,M,JUVW,KUVW)=0.D0
  205 CONTINUE
      S = S3
      T =   2.D0*S3
      AL(1,ID,2,ID,2)= 12.D0
      AL(1,ID,2,JD,1)= -T
      AL(1,ID,2,3 ,1)= -T
      AL(1,JD,1,ID,2)= -T
      AL(1,JD,1,JD,1)= 1.D0
      AL(1,JD,1,3 ,1)= 1.D0
      AL(4,JD,2,ID,2)= 12.D0
      AL(4,JD,2,JD,1)= -T
      AL(1,JD,2,JD,2)= 1.D0
      AL(2,JD,2,JD,2)= 12.D0
      AL(4,JD,2,3 ,1)= -T
      AL(1,JD,2,3 ,2)= 1.D0
      AL(2,JD,2,3 ,2)= 12.D0
      AL(1,3 ,1,ID,2)= -T
      AL(1,3 ,1,JD,1)= 1.D0
      AL(1,3 ,1,3 ,1)= 1.D0
      AL(4,3 ,2,ID,2)= 12.D0
      AL(4,3 ,2,JD,1)= -T
      AL(1,3 ,2,JD,2)= 1.D0
      AL(2,3 ,2,JD,2)= 12.D0
      AL(4,3 ,2,3 ,1)= -T
      AL(1,3 ,2,3 ,2)= 1.D0
      AL(2,3 ,2,3 ,2)= 12.D0
      DO 210 M=3,NTERMS
      AL(1,ID,2,ID,M)= -T*P(1,1,ID,M)
```

```
      AL(1,ID,2,JD,M)= -T*P(1,1,JD,M)                                    SS8J168
      AL(1,ID,2,3 ,M)= -T*P(1,1,3 ,M)                                    SS8J169
      AL(1,ID,M,ID,2)= -T*P(1,1,ID,M)                                    SS8J170
      AL(4,ID,M,ID,2)= -T*P(1,2,ID,M)                                    SS8J171
      AL(5,ID,M,ID,2)= -T*P(1,3,ID,M)                                    SS8J172
      AL(1,ID,M,JD,1)=    P(1,1,ID,M)                                    SS8J173
      AL(4,ID,M,JD,1)=    P(1,2,ID,M)                                    SS8J174
      AL(5,ID,M,JD,1)=    P(1,3,ID,M)                                    SS8J175
      AL(1,ID,M,JD,2)=  S*P(1,1,ID,M) -T*P(2,1,ID,M)                     SS8J176
      AL(2,ID,M,JD,2)= -T*P(1,2,ID,M)                                    SS8J177
      AL(4,ID,M,JD,2)=  S*P(1,2,ID,M) -T*P(2,2,ID,M)                     SS8J178
      AL(5,ID,M,JD,2)=  S*P(1,3,ID,M) -T*P(2,3,ID,M)                     SS8J179
      AL(6,ID,M,JD,2)= -T*P(1,3,ID,M)                                    SS8J180
      AL(1,ID,M,3 ,1)=    P(1,1,ID,M)                                    SS8J181
      AL(4,ID,M,3 ,1)=    P(1,2,ID,M)                                    SS8J182
      AL(5,ID,M,3 ,1)=    P(1,3,ID,M)                                    SS8J183
      AL(1,ID,M,3 ,2)=  S*P(1,1,ID,M) -T*P(2,1,ID,M)                     SS8J184
      AL(2,ID,M,3 ,2)= -T*P(1,2,ID,M)                                    SS8J185
      AL(4,ID,M,3 ,2)=  S*P(1,2,ID,M) -T*P(2,2,ID,M)                     SS8J186
      AL(5,ID,M,3 ,2)=  S*P(1,3,ID,M) -T*P(2,3,ID,M)                     SS8J187
      AL(6,ID,M,3 ,2)= -T*P(1,3,ID,M)                                    SS8J188
      AL(1,JD,1,ID,M)=    P(1,1,ID,M)                                    SS8J189
      AL(1,JD,1,JD,M)=    P(1,1,JD,M)                                    SS8J190
      AL(1,JD,1,3 ,M)=    P(1,1,3 ,M)                                    SS8J191
      AL(1,JD,2,ID,M)=  S*P(1,1,ID,M) -T*P(2,1,ID,M)                     SS8J192
      AL(2,JD,2,ID,M)= -T*P(1,2,ID,M)                                    SS8J193
      AL(4,JD,2,ID,M)= -T*P(1,1,ID,M)                                    SS8J194
      AL(1,JD,2,JD,M)=  S*P(1,1,JD,M) -T*P(2,1,JD,M)                     SS8J195
      AL(2,JD,2,JD,M)= -T*P(1,2,JD,M)                                    SS8J196
      AL(4,JD,2,JD,M)= -T*P(1,1,JD,M)                                    SS8J197
      AL(1,JD,2,3 ,M)=  S*P(1,1,3 ,M) -T*P(2,1,3 ,M)                     SS8J198
      AL(2,JD,2,3 ,M)= -T*P(1,2,3 ,M)                                    SS8J199
      AL(4,JD,2,3 ,M)= -T*P(1,1,3 ,M)                                    SS8J200
      AL(1,JD,M,ID,2)= -T*P(1,1,JD,M)                                    SS8J201
      AL(4,JD,M,ID,2)= -T*P(1,2,JD,M)                                    SS8J202
      AL(5,JD,M,ID,2)= -T*P(1,3,JD,M)                                    SS8J203
      AL(1,JD,M,JD,1)=    P(1,1,JD,M)                                    SS8J204
      AL(4,JD,M,JD,1)=    P(1,2,JD,M)                                    SS8J205
      AL(5,JD,M,JD,1)=    P(1,3,JD,M)                                    SS8J206
      AL(1,JD,M,JD,2)=  S*P(1,1,JD,M) -T*P(2,1,JD,M)                     SS8J207
      AL(2,JD,M,JD,2)= -T*P(1,2,JD,M)                                    SS8J208
      AL(4,JD,M,JD,2)=  S*P(1,2,JD,M) -T*P(2,2,JD,M)                     SS8J209
      AL(5,JD,M,JD,2)=  S*P(1,3,JD,M) -T*P(2,3,JD,M)                     SS8J210
      AL(6,JD,M,JD,2)= -T*P(1,3,JD,M)                                    SS8J211
      AL(1,JD,M,3 ,1)=    P(1,1,JD,M)                                    SS8J212
      AL(4,JD,M,3 ,1)=    P(1,2,JD,M)                                    SS8J213
      AL(5,JD,M,3 ,1)=    P(1,3,JD,M)                                    SS8J214
      AL(1,JD,M,3 ,2)=  S*P(1,1,JD,M) -T*P(2,1,JD,M)                     SS8J215
      AL(2,JD,M,3 ,2)= -T*P(1,2,JD,M)                                    SS8J216
      AL(4,JD,M,3 ,2)=  S*P(1,2,JD,M) -T*P(2,2,JD,M)                     SS8J217
      AL(5,JD,M,3 ,2)=  S*P(1,3,JD,M) -T*P(2,3,JD,M)                     SS8J218
      AL(6,JD,M,3 ,2)= -T*P(1,3,JD,M)                                    SS8J219
      AL(1,3 ,1,ID,M)=    P(1,1,ID,M)                                    SS8J220
      AL(1,3 ,1,JD,M)=    P(1,1,JD,M)                                    SS8J221
      AL(1,3 ,1,3 ,M)=    P(1,1,3 ,M)                                    SS8J222
      AL(1,3 ,2,ID,M)=  S*P(1,1,ID,M) -T*P(2,1,ID,M)                     SS8J223
```

```
      AL(2,3 ,2,ID,M)= -T*P(1,2,ID,M)
      AL(4,3 ,2,ID,M)= -T*P(1,1,ID,M)
      AL(1,3 ,2,JD,M)=  S*P(1,1,JD,M) -T*P(2,1,JD,M)
      AL(2,3 ,2,JD,M)= -T*P(1,2,JD,M)
      AL(4,3 ,2,JD,M)= -T*P(1,1,JD,M)
      AL(1,3 ,2,3 ,M)=  S*P(1,1,3 ,M) -T*P(2,1,3 ,M)
      AL(2,3 ,2,3 ,M)= -T*P(1,2,3 ,M)
      AL(4,3 ,2,3 ,M)= -T*P(1,1,3 ,M)
      AL(1,3 ,M,ID,2)= -T*P(1,1,3 ,M)
      AL(4,3 ,M,ID,2)= -T*P(1,2,3 ,M)
      AL(5,3 ,M,ID,2)= -T*P(1,3,3 ,M)
      AL(1,3 ,M,JD,1)=    P(1,1,3 ,M)
      AL(4,3 ,M,JD,1)=    P(1,2,3 ,M)
      AL(5,3 ,M,JD,1)=    P(1,3,3 ,M)
      AL(1,3 ,M,JD,2)=  S*P(1,1,3 ,M) -T*P(2,1,3 ,M)
      AL(2,3 ,M,JD,2)= -T*P(1,2,3 ,M)
      AL(4,3 ,M,JD,2)=  S*P(1,2,3 ,M) -T*P(2,2,3 ,M)
      AL(5,3 ,M,JD,2)=  S*P(1,3,3 ,M) -T*P(2,3,3 ,M)
      AL(6,3 ,M,JD,2)= -T*P(1,3,3 ,M)
      AL(1,3 ,M,3 ,1)=    P(1,1,3 ,M)
      AL(4,3 ,M,3 ,1)=    P(1,2,3 ,M)
      AL(5,3 ,M,3 ,1)=    P(1,3,3 ,M)
      AL(1,3 ,M,3 ,2)=  S*P(1,1,3 ,M) -T*P(2,1,3 ,M)
      AL(2,3 ,M,3 ,2)= -T*P(1,2,3 ,M)
      AL(4,3 ,M,3 ,2)=  S*P(1,2,3 ,M) -T*P(2,2,3 ,M)
      AL(5,3 ,M,3 ,2)=  S*P(1,3,3 ,M) -T*P(2,3,3 ,M)
      AL(6,3 ,M,3 ,2)= -T*P(1,3,3 ,M)
  210 CONTINUE
      DO 236 I=1,IPOWER
      T = I
      $W(I,ID,1,1,1) = 1./T
      $W(I,ID,2,1,1) = 0.
      $W(I,ID,3,1,1) = 0.
      $W(I,ID,1,1,2) = S3*(1./T-2./(T+1.))
      $W(I,ID,2,1,2) = 0.
      $W(I,ID,3,1,2) = 0.
      $W(I,ID,1,2,1) = S3*(1./T-2./(T+1.))
      $W(I,ID,2,2,1) = 0.
      $W(I,ID,3,2,1) = -2.*S3/T
      $W(I,ID,1,2,2) = 3.*(1./T-4./(T+1.)+4./(T+2.))
      $W(I,ID,2,2,2) = 12./T
      $W(I,ID,3,2,2) = -6.*(1./T-2./(T+1.))
      IF ( M.LE.2 )  GO TO 236
      DO 235 M=3,NTERMS
      $W(I,ID,1,1,M) = P(I,1,3,M)
      $W(I,ID,2,1,M) = 0.
      $W(I,ID,3,1,M) = 0.
      $W(I,ID,1,M,1) = P(I,1,3,M)
      $W(I,ID,2,M,1) = 0.
      $W(I,ID,3,M,1) = P(I,2,3,M)
      $W(I,ID,1,2,M) = S3*(P(I,1,3,M)-2.*P(I+1,1,3,M) )
      $W(I,ID,2,2,M) = -2.*S3*P(I,2,3,M)
      $W(I,ID,3,2,M) = -2.*S3*P(I,1,3,M)
      $W(I,ID,1,M,2) = S3*(P(I,1,3,M)-2.*P(I+1,1,3,M) )
      $W(I,ID,2,M,2) = -2.*S3*P(I,2,3,M)
  235 $W(I,ID,3,M,2) = S3*(P(I,2,3,M)-2.*P(I+1,2,3,M) )
```

```
  236 CONTINUE
  125 CONTINUE
C  CALCULATE MODE SHAPES AND DERIVATIVES
      DO 400 I=1,25
      W=I-1
      W=W/24.
      IF(MNIJ.NE.5) GO TO 300
      EVAL(1,3 ,1,I)= S3*W
      EVAL(2,3 ,1,I)= S3
      EVAL(1,JD,1,I)= S3*W
      EVAL(2,JD,1,I)= S3
      EVAL(1,ID,1,I)= S3
      EVAL(2,ID,1,I)= 0.D0
      GO TO 400
  300 EVAL(1,3 ,1,I)= 1.D0
      EVAL(2,3 ,1,I)= 0.D0
      EVAL(1,3 ,2,I)= S3*(1.D0-2.D0*W)
      EVAL(2,3 ,2,I)= -2.D0*S3
      EVAL(1,JD,1,I)= 1.D0
      EVAL(2,JD,1,I)= 0.D0
      EVAL(1,JD,2,I)= S3*(1.D0-2.D0*W)
      EVAL(2,JD,2,I)= -2.D0*S3
      EVAL(1,ID,1,I)= 0.
      EVAL(2,ID,1,I)= 0.D0
      EVAL(1,ID,2,I)= -2.D0*S3
      EVAL(2,ID,2,I)= 0.D0
  400 CONTINUE
      INNN=MNIJ-4
      DO 500 L=1,25
      DO 500 J=1,INNN
      DO 500 K=3,4
      DO 500 I=1,3
  500 EVAL(K,I,J,L) = 0.D0
      RETURN
      END
```

CC = 00315

```
      SUBROUTINE SEARCH ( KEY1, KEY2, M1,M2,MM,KM,LM,IM,NM,FMIN )
C **
C **  THIS SUBROUTINE KEEPS TRACK OF THE MINIMUM MARGIN OF SAFETY.
C **
      DIMENSION    F(15,25,25)
      COMMON / ARRAYS / F
C
      FH = FMIN
      DO 10 M=M1,M2
      DO 10 K=1,25
      DO 10 L=1,25
      IF (FH .LT. F(M,K,L)   )  GO TO 10
      FH = F(M,K,L)
      MH = M
      KH = K
      LH = L
   10 CONTINUE
      IF ( FMIN .LE. FH )  RETURN
      FMIN = FH
      MM = MH
      KM = KH
      LM = LH
      IM = KEY1
      NM = KEY2
      RETURN
      END
```

CC = 00026

```
      SUBROUTINE ASEMBL
C
C **  THIS SUBROUTINE ASSEMBLES THE POTENTIAL ENERGY MATRIX ( V ),
C **  THE KINETIC ENERGY MATRIX ( TT ), THE EDGE LOADS MATRIX ( U ),
C **  AND THE LATERAL LOADS VECTOR ( Q ).
C
      DIMENSION V(150,150), TT(150,150), VHOLD(150,150)
      DIMENSION U(50,50), Q(150), S(150)
      DIMENSION    QHOLD(150),     SHOLD(150)
      DIMENSION    AL(2,6,3,10,3,10),        EVAL(4,2,3,10,25),
     1             $W(10,2,3,10,10),         P(11,2,3,3,10)
      DIMENSION    A(3,3),         B(3,3),         D(3,3)
      DIMENSION    YBARS(100),     ZBARS(100),     AS(100),
     1             XIYYS(100),     XIYZS(100),     XIZZS(100),     ES(100),
     2             GJS(100),       RHOS(100),      PAXS(100),
     3             XBARR(50),      ZBARR(50),      AR(50),
     4             XIXXR(50),      XIXZR(50),      XIZZR(50),      ER(50),
     5             GJR(50),        RHOR(50),       PAXR(50),
     A             PMASS(50),      IPWW(50),       IPWY(50),
     B             PX(10,10),      PY(10,10),      PXY(10,10),
     C             PC(50),         IPXX(50),       IPYY(50),
     D             FC(50),         IFXX(50),       IFYY(50),
     E             ITAGCM(50),     QQ(10,10),
     F             PLMOM(50),      ITAGLM(50),     IDISLM(50),
     G             PKC(50),        IGSPRX(50),     IGSPRY(50),
     H             PLINE(50),      IDISLS(50),     ITAGLS(50)
      DIMENSION    ITIME(12),      TIME(50)
      DIMENSION    X(50),          Y(50)
C
      COMMON            U
      COMMON / BLOCK  / TT
      COMMON / ARRAYS / P,         AL,        $W
      COMMON / VALUES / EVAL
      COMMON / CNTROL / IFLAGD,    IFLAGB,    IFLAGW,    IBCX,      IBCY,
     1                  N1,  IEDGE,  IREACT,  N2(3),     IELAST,    INTPRT
      COMMON / NUMBER / NPLYS,     NTUX,      NTVX,      NTWX,      NTUY,
     1                  NTVY,      NTWY,      NMODES,    NSTRNG,    NRING,
     2                  NPNX,      NPNY,      NQTX,      NQTY,      NPTLDS,
     3                  NPTMOM,    NLNMOM,    NLMASS,    NPTSUP,    NLNSPR,
     4                  MATSIZ,   MUVSIZ,    MWSIZ
      COMMON / GEOM   / AA,        BB,        RR,        ALFAX,     ALFAY,
     1                  BETAX,     BETAY
      COMMON / $TIME  / TIME,      ITIME
      COMMON / ABD    / A,         B,         D,         RHAB
      COMMON / PARAM  / YBARS,     ZBARS,     AS,        XIYYS,     XIYZS,
     1                  XIZZS,     ES,        GJS,       RHOS,      PAXS,
     3                  XBARR,     ZBARR,     AR,        XIXXR,     XIXZR,
     4                  XIZZR,     ER,        GJR,       RHOR,      PAXR,
     6                  PMASS,     IPWW,      IPWY,      PX,        PY,
     7                  PXY,       PC,        IPXX,      IPYY,      FC,
     8                  IFXX,      IFYY,      ITAGCM,    QQ,        PLMOM,
     9                  ITAGLM,    IDISLM,    PKC,       IGSPRX,    IGSPRY,
     A                  PLINE,     IDISLS,    ITAGLS
      COMMON / STFVAL / ESV(10,100),    ESW(10,100),    ESDW(10,100),
     1                  ERU(10,50),     ERW(10,50),     ERDW(10,50)
      EQUIVALENCE  ( VHOLD(1),P(1) )
```

```
      EQUIVALENCE  ( QHOLD(1), YBARS(1) ), ( SHOLD(1), ZBARS(51) )
      DATA NAMEV/'V   '/,NAMETT/'TT  '/,NAMEU/'U   '/,NAMEQ/'Q   '/
      DATA NAMES / 'S   ' /
C
      ITHERY = 2
      IF ( INTPRT .NE. 1 )  GO TO 1001
      IF ( ITHERY .EQ. 1 )  WRITE (6,11)
   11 FORMAT ('0 USING NOVOZHILOV SHELL THEORY')
      IF ( ITHERY .EQ. 2 )  WRITE (6,12)
   12 FORMAT ('0 USING VLASOV SHELL THEORY')
      WRITE (6,4)
    4 FORMAT ('OTHE AL INTEGRALS FOLLOW')
      DO 990 I=1,2
      DO 990 I1=1,3
      DO 900 J1=1,3
      IF ( I .EQ. 2 )  GO TO 7
      IF ( I1 .EQ. 1 )  M1L = NTUX
      IF ( I1 .EQ. 2 )  M1L = NTVX
      IF ( I1 .EQ. 3 )  M1L = NTWX
      IF ( J1 .EQ. 1 )  M2L = NTUX
      IF ( J1 .EQ. 2 )  M2L = NTVX
      IF ( J1 .EQ. 3 )  M2L = NTWX
      GO TO 8
    7 IF ( I1 .EQ. 1 )  M1L = NTUY
      IF ( I1 .EQ. 2 )  M1L = NTVY
      IF ( I1 .EQ. 3 )  M1L = NTWY
      IF ( J1 .EQ. 1 )  M2L = NTUY
      IF ( J1 .EQ. 2 )  M2L = NTVY
      IF ( J1 .EQ. 3 )  M2L = NTWY
    8 CONTINUE
      DO 3 K1=1,6
      WRITE (6,1)  I,K1,I1,J1
    1 FORMAT ('0',4I2)
      DO 3 M1=1,M1L
      WRITE (6,2)  ( AL(I,K1,I1,M1,J1,M2), M2=1,M2L)
    2 FORMAT (' ',1P10E12.5)
    3 CONTINUE
  900 CONTINUE
      DO 930 ID1=1,4
      WRITE (6,931) ID1,I,I1
  931 FORMAT ('OEVAL ',3I2)
      DO 930 LL=1,25
  930 WRITE (6,2) (EVAL(ID1,I,I1,M1,LL), M1=1,M1L )
      IF (I.EQ.1) MAXP = MAX0 (NPNX,NQTX,1)
      IF (I.EQ.2) MAXP = MAX0 (NPNY,NQTY,1)
      DO 940 IP=1,MAXP
      DO 940 K2=1,3
      WRITE(6,941) IP,I,K2,I1
  941 FORMAT ('OP INTEGRALS ',4I2)
  940 WRITE (6,2) ( P(IP,I,K2,I1,M1), M1=1,M1L )
  990 CONTINUE
      WRITE (6,901)
  901 FORMAT ('OTHE W**2 INTEGRALS FOLLOW')
      DO 920 I=1,2
      IF (I.EQ.2) GO TO 902
      MAXP = MAX0 (NPNX,NQTX,1)
```

```
      M3L = NTWX
      L3L = NTWX
      GO TO 903
  902 MAXP = MAXO (NPNY,NQTY,1)
      M3L = NTWY
      L3L = NTWY
  903 DO 920 IP=1,MAXP
      DO 920 K2=1,3
      WRITE (6,1) IP,I,K2
      DO 920 L3=1,L3L
  920 WRITE (6,2)  ( $W(IP,I,K2,L3,M3),M3=1,M3L)
 1001 CONTINUE
      DO 5 I=1,50
      X(I)=0.
    5 Y(I)=0.
      DO 6 I=1,MWSIZ
      DO 6 J=1,MWSIZ
    6 U(I,J) = 0.
      DO 10 I = 1,MATSIZ
      Q(I) = 0.
      S(I) = 0.
      DO 10 J = 1,MATSIZ
      V(I,J) = 0.
      TT(I,J) = 0.
   10 CONTINUE
      L = 1
      K = 1
      A1     = 1./AA
      B1     = 1./BB
      R1     = 1./RR
      A1B    = A1*BB
      AB1    = AA*B1
      A1BR1  = A1B*R1
      AB1R1  = AB1*R1
      BR1    = BB*R1
      AR1    = AA*R1
      A2B    = A1B*A1
      AB2    = AB1*B1
      B1R2   = B1*R1*R1
      R2     = R1*R1
      BR2    = BR1*R1
      AR2    = AR1*R1
      A1R1   = A1*R1
      A2BR1  = A2B*R1
      AB2R1  = AB2*R1
      B1R1   = B1*R1
      TBR2   = 2.*BR2
      ABR2   = AA*BR2
      A3B    = A2B/AA
      A1B1   = A1*B1
      A2     = A1*A1
      AB3    = AB2/BB
      B2     = B1*B1
      B3 = B2*B1
      AB = AA*BB
      A1B1R1 = A1*B1R1
```

```
      A1B2 = A1B1*B1
      A2B1 = A1*A1B1
      A3   = A1*A2
      AB1R2 = AB1*R2
      A1BR2 = A1B*R2
      A2BR2 = A2B*R2
      ABR1 = AA*BR1
      AR3 = AR2 * R1
      BR3 = BR2 * R1
      ABR3 = ABR2 * R1
      ABR4 = ABR3 * R1
      CALL STATUS ( ITIME )
      TIME(5) = .01*ITIME(8)
      DO 1000  IP = 1,3
      DO 1000  IQ = 1,3
      IF ( IP .EQ. 1 )  NTLI = NTUX
      IF ( IP .EQ. 1 )  NTLJ = NTUY
      IF ( IP .EQ. 2 )  NTLI = NTVX
      IF ( IP .EQ. 2 )  NTLJ = NTVY
      IF ( IP .EQ. 3 )  NTLI = NTWX
      IF ( IP .EQ. 3 )  NTLJ = NTWY
      IF ( IQ .EQ. 1 )  NTLM = NTUX
      IF ( IQ .EQ. 1 )  NTLN = NTUY
      IF ( IQ .EQ. 2 )  NTLM = NTVX
      IF ( IQ .EQ. 2 )  NTLN = NTVY
      IF ( IQ .EQ. 3 )  NTLM = NTWX
      IF ( IQ .EQ. 3 )  NTLN = NTWY
      DO 1000  I = 1,NTLI
      DO 1000  J = 1,NTLJ
      DO 1000  M = 1,NTLM
      DO 1000  N = 1,NTLN
      IF ( IP .EQ. 1 )  II = (I-1)*NTUY + J
      IF ( IP .EQ. 2 )  II = NTUX*NTUY + (I-1)*NTVY + J
      IF ( IP .EQ. 3 )  II = NTUX*NTUY + NTVX*NTVY + (I-1)*NTWY + J
      IF ( IQ .EQ. 1 )  JJ = (M-1)*NTUY + N
      IF ( IQ .EQ. 2 )  JJ = NTUX*NTUY + (M-1)*NTVY + N
      IF ( IQ .EQ. 3 )  JJ = NTUX*NTUY + NTVX*NTVY + (M-1)*NTWY + N
      KK = II -MUVSIZ
      LL = JJ -MUVSIZ
      IF ( IP .GT. IQ )  GO TO 580
      IF ( IP .EQ. 1 .AND. IQ .EQ. 1 )  GO TO 20
      IF ( IP .EQ. 1 .AND. IQ .EQ. 2 )  GO TO 100
      IF ( IP .EQ. 1 .AND. IQ .EQ. 3 )  GO TO 160
      IF ( IP .EQ. 2 .AND. IQ .EQ. 2 )  GO TO 220
      IF ( IP .EQ. 2 .AND. IQ .EQ. 3 )  GO TO 310
      IF ( IP .EQ. 3 .AND. IQ .EQ. 3 )  GO TO 370
   20 X( 1) =                          AL(1,2,1,I,1,M) * AL(2,1,1,J,1,N)
      X( 2) =                          AL(1,4,1,I,1,M) * AL(2,4,1,N,1,J)
      X( 3) =                          AL(1,4,1,M,1,I) * AL(2,4,1,J,1,N)
      X( 4) =                          AL(1,1,1,I,1,M) * AL(2,2,1,J,1,N)
      Y( 1) = A(1,1) * A1B * X( 1) + A(1,3) * ( X( 2) + X( 3) )
     1       + A(3,3) * AB1 * X( 4)
      V(II,JJ) = V(II,JJ) + Y( 1)
      IF ( ITHERY .EQ. 2 )  V(II,JJ) = V(II,JJ) - B(1,3) * R1 * (X(2)
     1  + X(3)) - 2.* B(3,3) * AB1R1 * X(4) + D(3,3) * AB1R2 * X(4)
      IF ( NSTRNG .EQ. 0 )  GO TO 30
```

```
      DO 30 L=1,NSTRNG
      V(II,JJ) = V(II,JJ) + A1 * ES(L) * AS(L) *   AL(1,2,1,I,1,M)
     1            * ESW(J,L) * ESW(N,L)
   30 CONTINUE
      IF ( NRING .EQ. 0 )  GO TO 40
      DO 40 K=1,NRING
      V(II,JJ) = V(II,JJ) + B3 * ER(K) * XIZZR(K) *   AL(2,3,1,J,1,N)
     1            * ERU(I,K) * ERU(M,K)
   40 CONTINUE
      IF ( IFLAGD .EQ. 0 )  GO TO 70
      TT(II,JJ) =           RHAB *   AL(1,1,1,I,1,M) *   AL(2,1,1,J,1,N)
      IF ( NSTRNG .EQ. 0 )  GO TO 50
      DO 50 L=1,NSTRNG
      TT(II,JJ) = TT(II,JJ) + RHOS(L) * AS(L) *   AL(1,1,1,I,1,M) * AA
     1            * ESW(J,L) * ESW(N,L)
   50 CONTINUE
      IF ( NRING .EQ. 0 )  GO TO 60
      DO 60 K=1,NRING
      TT(II,JJ) = TT(II,JJ) + RHOR(K) * ( BB * AR(K) *   AL(2,1,1,J,1,N)
     1             + XIZZR(K) * B1 *   AL(2,2,1,J,1,N) )
     2            * ERU(I,K) * ERU(M,K)
   60 CONTINUE
      IF ( NLMASS .EQ. 0 )  GO TO 70
      DO 70 L=1,NLMASS
      TT(II,JJ) = TT(II,JJ) + PMASS(L) * EVAL(1,1,1,I,IPWW(L)) *
     1EVAL(1,2,1,J,IPWY(L))*EVAL(1,1,1,M,IPWW(L))*EVAL(1,2,1,N,IPWY(L))
   70 CONTINUE
      IF ( IFLAGW .EQ. 0 )  GO TO 1000
      IF ( JJ     .GT. 1 )  GO TO 1000
      IF ( IEDGE  .EQ. 0 )  GO TO 75
      IF ( NSTRNG .EQ. 0 )  GO TO 72
      DO 72 L=1,NSTRNG
      S(II) = S(II) - PAXS(L) * P(1,1,2,1,I) * ESW(J,L)
   72 CONTINUE
      IF ( NRING  .EQ. 0 )  GO TO 73
      DO 73 K=1,NRING
      S(II) = S(II) - PAXR(K) * XBARR(K) * P(1,2,3,1,J) * ERU(I,K)
   73 CONTINUE
      DO 74 K=1,NPNX
      DO 74 L=1,NPNY
   74 S(II) = S(II) - BB * PX (K,L) * P(K,1,2,1,I) * P(L,2,1,1,J)
     1                - AA * PXY(K,L) * P(K,1,1,1,I) * P(L,2,2,1,J)
   75 IF ( NPTMOM .EQ. 0 )  GO TO   80
      DO 80 L=1,NPTMOM
      IF ( ITAGCM(L) .EQ. 1 )  GO TO 80
      Q(II) = Q(II) - R1 * FC(L) * EVAL(1,1,1,I,IFXX(L))
     1                 * EVAL(1,2,1,J,IFYY(L))
   80 CONTINUE
      IF ( NLNMOM .EQ. 0 )  GO TO 1000
      DO 90 L=1,NLNMOM
      IF ( ITAGLM(L) .EQ. 1 )  GO TO 90
      Q(II) = Q(II) - BR1 * PLMOM(L) * EVAL(1,1,1,I,IDISLM(L))
     1                 * P(1,2,1,1,J)
   90 CONTINUE
      GO TO 1000
  100 X( 5) =                          AL(1,4,1,I,2,M) * AL(2,4,2,N,1,J)
```

```
      X( 6) =                         AL(1,2,1,I,2,M) * AL(2,1,1,J,2,N)
      X( 7) =                         AL(1,1,1,I,2,M) * AL(2,2,1,J,2,N)
      X( 8) =                         AL(1,4,2,M,1,I) * AL(2,4,1,J,2,N)
      Y( 2) = A(1,2) *        X( 5) +      A(1,3) * A1B    * X( 6)
     1      + A(2,3) * AB1    * X( 7) +      A(3,3)          * X( 8)
      IF ( ITHERY .NE. 1 )  GO TO 105
      Y( 3) = B(1,2) * R1     * X( 5) + 2. * B(1,3) * A1BR1 * X( 6)
     1      + B(2,3) * AB1R1 * X( 7) + 2. * B(3,3) * R1     * X( 8)
      GO TO 110
  105 Y(3) = B(1,3) * A1BR1 * X(6) - B(2,3) * AB1R1 * X(7)
     1      - D(3,3) * R2 * X(8)
  110 V(II,JJ) = V(II,JJ) + Y(2) + Y(3)
      IF ( NSTRNG .EQ. 0 )  GO TO 120
      DO 120 L=1,NSTRNG
      V(II,JJ) = V(II,JJ) - A2 * ES(L) * AS(L) * YBARS(L)
     1          * AL(1,6,2,M,1,I) * ESW(J,L) * ESV(N,L)
  120 CONTINUE
      IF ( NRING  .EQ. 0 )  GO TO 130
      DO 130 K=1,NRING
      V(II,JJ) = V(II,JJ) - B2 * ER(K) * AR(K) * XBARR(K)
     2          * AL(2,6,1,J,2,N) * ERU(I,K) * ERW(M,K)
  130 CONTINUE
      IF ( IFLAGD .EQ. 0 )  GO TO 1000
      IF ( NSTRNG .EQ. 0 )  GO TO  140
      DO 140 L=1,NSTRNG
      TT(II,JJ) = TT(II,JJ) - RHOS(L) * AS(L) * YBARS(L)
     1           * AL(1,4,2,M,1,I) * ESW(J,L) * ESV(N,L)
  140 CONTINUE
      IF ( NRING  .EQ. 0 )  GO TO 1000
      DO 150 K=1,NRING
      TT(II,JJ) = TT(II,JJ) - RHOR(K) * AR(K) * XBARR(K)
     2           * AL(2,4,1,J,2,N) * ERU(I,K) *ERW(M,K)
  150 CONTINUE
      GO TO 1000
  160 X( 9) =                         AL(1,4,1,I,3,M) * AL(2,1,1,J,3,N)
      X(10) =                         AL(1,1,1,I,3,M) * AL(2,4,1,J,3,N)
      X(11) =                         AL(1,6,3,M,1,I) * AL(2,1,1,J,3,N)
      X(12) =                         AL(1,4,1,I,3,M) * AL(2,5,3,N,1,J)
      X(13) =                         AL(1,2,1,I,3,M) * AL(2,4,3,N,1,J)
      X(14) =                         AL(1,5,3,M,1,I) * AL(2,4,1,J,3,N)
      X(15) =                         AL(1,1,1,I,3,M) * AL(2,6,3,N,1,J)
      X(16) =                         AL(1,4,3,M,1,I) * AL(2,2,1,J,3,N)
      Y( 4) = A(1,2) * BR1    * X( 9) +      A(2,3) * AR1    * X(10)
      IF ( ITHERY .NE. 1 )  GO TO 165
      Y( 5) =-B(1,1) * A2B    * X(11) -      B(1,2) * B1     * X(12)
     1      - B(1,3) * A1 * ( 2. * X(13) + X(14) )
     2      - B(2,3) * AB2    * X(15) - 2. * B(3,3) * B1     * X(16)
      GO TO 170
  165 Y(5) = - B(1,1) * A2B * X(11) - B(1,2) * ( BR2 * X(9) + B1 *X(12))
     1       - B(1,3) * A1 * ( 2.*X(13) + X(14) ) - B(2,3) * ( 2. * AR2
     2       * X(10) + AB2 * X(15) ) - 2.* B(3,3) * B1 * X(16)
     3       + D(1,3) * A1R1 * X(14) + D(2,3) * ( AR3 * X(10)
     4       + AB2R1 * X(15) ) + 2.* D(3,3) * B1R1 * X(16)
  170 V(II,JJ) = V(II,JJ) + Y(4) + Y(5)
      IF ( NSTRNG .EQ. 0 )  GO TO 180
      DO 180 L=1,NSTRNG
```

```
      V(II,JJ) = V(II,JJ) - ES(L) * AS(L) * ZBARS(L) * A2
     1          * AL(1,6,3,M,1,I) * ESW(J,L) * ESW(N,L)
  180 CONTINUE
      IF ( NRING  .EQ. 0 )  GO TO 190
      DO 190 K=1,NRING
      V(II,JJ) = V(II,JJ) + ERU(I,K) * ER(K) * ( ERW(M,K) * ( XIXZR(K)
     1           * B3 * AL(2,3,1,J,3,N) - AR(K) * XBARR(K) * B1R1
     2           * AL(2,5,1,J,3,N) ) - XIZZR(K) * A1B1R1 * ERDW(M,K)
     3           * AL(2,5,1,J,3,N) )
  190 CONTINUE
      IF ( IFLAGD .EQ. 0 )  GO TO 1000
      IF ( NSTRNG .EQ. 0 )  GO TO  200
      DO 200 L=1,NSTRNG
      TT(II,JJ) = TT(II,JJ) - RHOS(L) * AS(L) * ZBARS(L)
     1          * AL(1,4,3,M,1,I) * ESW(J,L) * ESW(N,L)
  200 CONTINUE
      IF ( NRING  .EQ. 0 )  GO TO 1000
      DO 210 K=1,NRING
      TT(II,JJ) = TT(II,JJ) + RHOR(K) * ERU(I,K) * ( -ZBARR(K) * A1B
     1          * AR(K) * ERDW(M,K) * AL(2,1,1,J,3,N) + B1 * XIXZR(K)
     2          * AL(2,2,1,J,3,N) * ERW(M,K) )
  210 CONTINUE
      GO TO 1000
  220 X(17) =                     AL(1,1,2,I,2,M) * AL(2,2,2,J,2,N)
      X(18) =                     AL(1,4,2,I,2,M) * AL(2,4,2,N,2,J)
      X(19) =                     AL(1,4,2,M,2,I) * AL(2,4,2,J,2,N)
      X(20) =                     AL(1,2,2,I,2,M) * AL(2,1,2,J,2,N)
      Y(6) = A(2,2) * AB1   * X(17) +      A(2,3) * ( X(18) + X(19) )
     1     + A(3,3) * A1B   * X(20)
      IF ( ITHERY .NE. 1 )  GO TO 225
      Y(7) = 2. * B(2,2) * AB1R1 * X(17) + 3. * B(2,3) * R1 * ( X(18)
     1     + X(19) ) + 4. * B(3,3) * A1BR1 * X(20)
      Y(8) = D(2,2) * AB1R2 * X(17) + 2. * D(2,3) * R2 * ( X(18)+X(19) )
     1     + 4. * D(3,3) * A1BR2 * X(20)
      GO TO 230
  225 Y(7) = B(2,3) * R1 * ( X(18) + X(19) ) + 2.* B(3,3)*A1BR1*X(20)
      Y(8) = D(3,3) * A1BR2 * X(20)
  230 V(II,JJ) = V(II,JJ) + Y(6) + Y(7) + Y(8)
      IF ( NSTRNG .EQ. 0 )  GO TO 240
      DO 240 L=1,NSTRNG
      V(II,JJ) = V(II,JJ) + ES(L) * XIZZS(L) * A3 * AL(1,3,2,I,2,M)
     1           * ESV(J,L) * ESV(N,L)
  240 CONTINUE
      IF ( NRING  .EQ. 0 )  GO TO 250
      DO 250 K=1,NRING
      V(II,JJ) = V(II,JJ) +ER(K) * AR(K) * B1 * AL(2,2,2,J,2,N)
     1           * ERW(I,K) * ERW(M,K)
  250 CONTINUE
      IF ( IFLAGD .EQ. 0 )  GO TO 280
      TT(II,JJ) = TT(II,JJ) + RHAB * AL(1,1,2,I,2,M) * AL(2,1,2,J,2,N)
      IF ( NSTRNG .EQ. 0 )  GO TO 260
      DO 260 L=1,NSTRNG
      TT(II,JJ) = TT(II,JJ) + RHOS(L) * ESV(J,L)*ESV(N,L)*(AA*AS(L)
     2          * AL(1,1,2,I,2,M) + A1 * XIZZS(L) * AL(1,2,2,I,2,M) )
  260 CONTINUE
      IF ( NRING  .EQ. 0 )  GO TO 270
```

```
      DO 270 K=1,NRING
      TT(II,JJ) = TT(II,JJ) + RHOR(K) * AR(K) * BB * AL(2,1,2,J,2,N)
     1          * ERW(I,K) * ERW(M,K)
  270 CONTINUE
      IF ( NLMASS .EQ. 0 )  GO TO 280
      DO 280 K=1,NLMASS
      TT(II,JJ) = TT(II,JJ) + PMASS(K) * EVAL(1,1,2,I,IPWW(K)) *
     1EVAL(1,2,2,J,IPWY(K))*EVAL(1,1,2,M,IPWW(K))*EVAL(1,2,2,N,IPWY(K))
  280 CONTINUE
      IF ( IFLAGW .EQ. 0 )  GO TO 1000
      IF ( JJ .GT. NTUX*NTUY + 1 )  GO TO 1000
      IF ( IEDGE  .EQ. 0 )  GO TO 285
      IF ( NSTRNG .EQ. 0 )  GO TO 282
      DO 282 L=1,NSTRNG
      S(II) = S(II) + PAXS(L) * A1 * YBARS(L) * P(1,1,3,2,I)*ESV(J,L)
  282 CONTINUE
      IF ( NRING  .EQ. 0 )  GO TO 283
      DO 283 K=1,NRING
      S(II) = S(II) - PAXR(K) * P(1,2,2,2,J) * ERW(I,K)
  283 CONTINUE
      DO 284 K=1,NPNX
      DO 284 L=1,NPNY
  284 S(II) = S(II) - AA * PY (K,L) * P(K,1,1,2,I) * P(L,2,2,2,J)
     1              - BB * PXY(K,L) * P(K,1,2,2,I) * P(L,2,1,2,J)
  285 IF ( NPTMOM .EQ. 0 )  GO TO 290
      DO 290 K=1,NPTMOM
      IF ( ITAGCM(K) .EQ. 2 )  GO TO 290
      Q(II) = Q(II) - R1 * FC(K) * EVAL(1,1,2,I,IFXX(K))
     1      * EVAL(1,2,2,J,IFYY(K))
  290 CONTINUE
      IF ( NLNMOM .EQ. 0 ) GO TO 1000
      DO 300 K=1,NLNMOM
      IF ( ITAGLM(K) .EQ. 2 )  GO TO 300
      Q(II) = Q(II) - AR1 * PLMOM(K) * EVAL(1,2,2,J,IDISLM(K))
     1              * P(1,1,1,2,I)
  300 CONTINUE
      GO TO 1000
  310 X(21) =                      AL(1,1,2,I,3,M) * AL(2,4,2,J,3,N)
      X(22) =                      AL(1,4,2,I,3,M) * AL(2,1,2,J,3,N)
      X(23) =                      AL(1,5,3,M,2,I) * AL(2,4,2,J,3,N)
      X(24) =                      AL(1,6,3,M,2,I) * AL(2,1,2,J,3,N)
      X(25) =                      AL(1,4,3,M,2,I) * AL(2,2,2,J,3,N)
      X(26) =                      AL(1,4,2,I,3,M) * AL(2,5,3,N,2,J)
      X(27) =                      AL(1,2,2,I,3,M) * AL(2,4,3,N,2,J)
      X(28) =                      AL(1,1,2,I,3,M) * AL(2,6,3,N,2,J)
      Y( 9) = A(2,2) * AR1 * X(21) + A(2,3) * BR1 * X(22)
      IF ( ITHERY .NE. 1 )  GO TO 315
      Y(10) = - B(1,2) * A1 * X(23)- B(1,3) * A2B * X(24)
     1       + B(2,2) *(AR2 * X(21) - AB2 * X(28) )
     2       + B(2,3) * ( TBR2 * X(22) - 2. * B1 * X(25) - B1 * X(26) )
     3       - B(3,3) * 2. * A1 * X(27)
      Y(11) = - D(1,2) * A1R1 * X(23) - 2. * D(1,3) * A2BR1 * X(24)
     1         - D(2,2) * AB2R1* X(28) - 2. * D(2,3) * B1R1 *( X(26)
     2         + X(25)) - 4. * D(3,3) * A1R1 * X(27)
      GO TO 320
  315 Y(10) = - B(1,2) *A1 * X(23) - B(1,3) * A2B * X(24)
```

```
     1         - B(2,2) * ( AR2 * X(21) + AB2 * X(28) )                SS8L448
     2         - B(2,3) * B1 * ( 2.*X(25) + X(26) )                    SS8L449
     3         - 2.*B(3,3) * A1 * X(27)                                SS8L450
      Y(11) = - D(1,3) * A2BR1 * X(24) - D(2,3) * ( B1R1 * X(26)       SS8L451
     1        + BR3 * X(22) ) - 2.*D(3,3) * A1R1 * X(27)               SS8L452
  320 V(II,JJ) = V(II,JJ) + Y(9) + Y(10) + Y(11)                       SS8L453
      IF ( NSTRNG .EQ. 0 )  GO TO 330                                  SS8L454
      DO 330 L=1,NSTRNG                                                SS8L455
      V(II,JJ) = V(II,JJ) + ES(L) * XIYZS(L) * A3 * AL(1,3,2,I,3,M)    SS8L456
     1         * ESV(J,L) * ESW(N,L)                                   SS8L457
  330 CONTINUE                                                         SS8L458
      IF ( NRING  .EQ. 0 )  GO TO 340                                  SS8L459
      DO 340 K=1,NRING                                                 SS8L460
      V(II,JJ) = V(II,JJ) + ER(K) * AR(K) * ERW(I,K)                   SS8L461
     1         * ( ERW(M,K)                * (-ZBARR(K) * B2           SS8L462
     2         * AL(2,6,3,N,2,J) + R1 * AL(2,4,2,J,3,N) )  + XBARR(K)  SS8L463
     3         * A1R1 * ERDW(M,K)                 * AL(2,4,2,J,3,N) )  SS8L464
  340 CONTINUE                                                         SS8L465
      IF ( IFLAGD .EQ. 0 )  GO TO 1000                                 SS8L466
      IF ( NSTRNG .EQ. 0 )  GO TO 350                                  SS8L467
      DO 350 L=1,NSTRNG                                                SS8L468
      TT(II,JJ) = TT(II,JJ) + RHOS(L) * ESV(J,L)                       SS8L469
     1          * ( - AB1 * ZBARS(L) * AS(L) * AL(1,1,2,I,3,M)         SS8L470
     2          * ESDW(N,L)                   + A1 * XIYYS(L)          SS8L471
     3          * AL(1,2,2,I,3,M) * ESW(N,L) )                         SS8L472
  350 CONTINUE                                                         SS8L473
      IF ( NRING .EQ. 0 )  GO TO 1000                                  SS8L474
      DO 360 K=1,NRING                                                 SS8L475
      TT(II,JJ) = TT(II,JJ) - RHOR(K) * AR(K) * ZBARR(K)               SS8L476
     2          * AL(2,4,3,N,2,J) * ERW(I,K) * ERW(M,K)                SS8L477
  360 CONTINUE                                                         SS8L478
      GO TO 1000                                                       SS8L479
  370 X(29) =                         AL(1,1,3,I,3,M) * AL(2,1,3,J,3,N)  SS8L480
      X(30) =                         AL(1,5,3,I,3,M) * AL(2,1,3,J,3,N)  SS8L481
      X(31) =                         AL(1,5,3,M,3,I) * AL(2,1,3,J,3,N)  SS8L482
      X(32) =                         AL(1,1,3,I,3,M) * AL(2,5,3,N,3,J)  SS8L483
      X(33) =                         AL(1,1,3,I,3,M) * AL(2,5,3,J,3,N)  SS8L484
      X(34) =                         AL(1,4,3,M,3,I) * AL(2,4,3,N,3,J)  SS8L485
      X(35) =                         AL(1,4,3,I,3,M) * AL(2,4,3,J,3,N)  SS8L486
      X(36) =                         AL(1,3,3,I,3,M) * AL(2,1,3,J,3,N)  SS8L487
      X(37) =                         AL(1,5,3,M,3,I) * AL(2,5,3,J,3,N)  SS8L488
      X(38) =                         AL(1,5,3,I,3,M) * AL(2,5,3,N,3,J)  SS8L489
      X(39) =                         AL(1,6,3,M,3,I) * AL(2,4,3,J,3,N)  SS8L490
      X(40) =                         AL(1,6,3,I,3,M) * AL(2,4,3,N,3,J)  SS8L491
      X(41) =                         AL(1,1,3,I,3,M) * AL(2,3,3,J,3,N)  SS8L492
      X(42) =                         AL(1,4,3,M,3,I) * AL(2,6,3,J,3,N)  SS8L493
      X(43) =                         AL(1,4,3,I,3,M) * AL(2,6,3,N,3,J)  SS8L494
      X(44) =                         AL(1,2,3,I,3,M) * AL(2,2,3,J,3,N)  SS8L495
      Y(12) = A(2,2) * ABR2 * X(29)                                    SS8L496
      IF ( ITHERY .NE. 1 )  GO TO 375                                  SS8L497
      Y(13) =-B(1,2) * A1BR1 * ( X(30) + X(31) )                       SS8L498
     1       - B(2,2) * AB1R1 * ( X(32) + X(33) )                      SS8L499
     2       - B(2,3) * 2.*R1 * ( X(34) + X(35) )                      SS8L500
      Y(14) = D(1,1) * A3B   * X(36) + D(1,2) * A1B1  * ( X(37)+X(38) ) SS8L501
     1       + D(1,3) * 2.*A2 * ( X(39)+X(40) ) + D(2,2) * AB3 * X(41)  SS8L502
     2       + D(2,3) * 2.*B2 * ( X(42)+X(43) ) + D(3,3) * 4.*A1B1* X(44)SS8L503
```

```
      GO TO 379                                                         SS8L504
  375 Y(13) = - B(1,2) * A1BR1 * ( X(30) + X(31) ) - B(2,2) * AB1R1 *   SS8L505
     1          ( X(32) + X(33) ) - 2.*B(2,2) * ABR3 * X(29)            SS8L506
     2          - 2.*B(2,3)* R1 * ( X(34) + X(35) )                     SS8L507
      Y(14) = D(1,1) * A3B * X(36) + D(1,2) * A1B1 * ( X(37) + X(38) )  SS8L508
     1       + D(1,2) * A1BR2 * ( X(30) + X(31) ) + 2.*D(1,3) * A2 *    SS8L509
     2       ( X(39) + X(40) ) + D(2,2) *(AB3 * X(41) + ABR4 * X(29)    SS8L510
     3       + AB1R2 * ( X(32) + X(33) ) ) + 2.*D(2,3) * ( B2 * ( X(42) SS8L511
     4       + X(43) ) + R2 * ( X(34) + X(35) ) ) + 4.*D(3,3)*A1B1*X(44) SS8L512
  379 V(II,JJ) = V(II,JJ) + Y(12) + Y(13) + Y(14)                       SS8L513
      IF ( NSTRNG .EQ. 0 )  GO TO 380                                   SS8L514
      DO 380 L=1,NSTRNG                                                 SS8L515
      V(II,JJ) = V(II,JJ) + ES(L) * XIYYS(L) * A3 * AL(1,3,3,I,3,M)     SS8L516
     1           * ESW(J,L) * ESW(N,L)                                  SS8L517
     2           + GJS(L) * A1B2 * AL(1,2,3,I,3,M)                      SS8L518
     3           * ESDW(J,L) * ESDW(N,L)                                SS8L519
  380 CONTINUE                                                          SS8L520
      IF ( NRING  .EQ. 0 )  GO TO 390                                   SS8L521
      DO 390 K=1,NRING                                                  SS8L522
      V(II,JJ) = V(II,JJ) + ER(K) * XIXXR(K) * B3 * AL(2,3,3,J,3,N)     SS8L523
     1           * ERW(I,K) * ERW(M,K)                                  SS8L524
     2           + GJR(K) * A2B1 * AL(2,2,3,J,3,N)                      SS8L525
     3           * ERDW(I,K) * ERDW(M,K)                                SS8L526
  390 CONTINUE                                                          SS8L527
      IF ( IELAST .EQ. 1 )  GO TO 400                                   SS8L528
      V(II,JJ) = V(II,JJ) + A3B * D(1,1) * AL(2,1,3,J,3,N) *            SS8L529
     1           ( ALFAX * EVAL(2,1,3,I,1) * EVAL(2,1,3,M,1)            SS8L530
     2           + BETAX * EVAL(2,1,3,I,25)* EVAL(2,1,3,M,25) )         SS8L531
     3           + AB3 * D(2,2) * AL(1,1,3,I,3,M) *                     SS8L532
     4           ( ALFAY * EVAL(2,2,3,J,1) * EVAL(2,2,3,N,1)            SS8L533
     5           + BETAY * EVAL(2,2,3,J,25)* EVAL(2,2,3,N,25) )         SS8L534
  400 CONTINUE                                                          SS8L535
      IF ( NPTSUP .EQ. 0 )  GO TO 410                                   SS8L536
      DO 410 L=1,NPTSUP                                                 SS8L537
      V(II,JJ) = V(II,JJ) + PKC(L)                                      SS8L538
     1           * EVAL(1,1,3,I,IGSPRX(L)) * EVAL(1,1,3,M,IGSPRX(L))    SS8L539
     2           * EVAL(1,2,3,J,IGSPRY(L)) * EVAL(1,2,3,N,IGSPRY(L))    SS8L540
  410 CONTINUE                                                          SS8L541
      IF ( NLNSPR .EQ. 0 )  GO TO 430                                   SS8L542
      DO 430 L=1,NLNSPR                                                 SS8L543
      IF ( ITAGLS(L) .EQ. 2 )  GO TO 420                                SS8L544
      V(II,JJ) = V(II,JJ) + PLINE(L) * AA * AL(1,1,3,I,3,M)             SS8L545
     1           * EVAL(1,2,3,J,IDISLS(L)) * EVAL(1,2,3,N,IDISLS(L))    SS8L546
      GO TO 430                                                         SS8L547
  420 V(II,JJ) = V(II,JJ) + PLINE(L) * BB * AL(2,1,3,J,3,N)             SS8L548
     1           * EVAL(1,1,3,I,IDISLS(L)) * EVAL(1,1,3,M,IDISLS(L))    SS8L549
  430 CONTINUE                                                          SS8L550
      IF ( NRING  .EQ. 0 )  GO TO 450                                   SS8L551
      DO 450 K=1,NRING                                                  SS8L552
      V(II,JJ) = V(II,JJ) + ER(K) * ( ERW(I,K)        * ( AR(K)         SS8L553
     1           * ERW(M,K)        * ( BR2 * AL(2,1,3,J,3,N)            SS8L554
     2           - ZBARR(K) * B1R1 * ( AL(2,5,3,J,3,N) + AL(2,5,3,N,3,J)))SS8L555
     3           +  ERDW(M,K)      * ( - XIXZR(K) * A1B1R1*AL(2,5,3,J,3,N)SS8L556
     4           + AR(K) * XBARR(K) * A1BR2 * AL(2,1,3,J,3,N) ) )       SS8L557
     5           + ERDW(I,K) * ( ERW(M,K)* ( - XIXZR(K)                 SS8L558
     6           * A1B1R1 * AL(2,5,3,N,3,J) + AR(K) * XBARR(K) * A1BR2  SS8L559
```

```
     7          * AL(2,1,3,J,3,N) ) + ERDW(M,K)        * XIZZR(K)        SS8L560
     8          * A2BR2 * AL(2,1,3,J,3,N) ) )                            SS8L561
  450 CONTINUE                                                           SS8L562
      IF ( IFLAGD .EQ. 0 )  GO TO 480                                    SS8L563
      TT(II,JJ) = TT(II,JJ) + RHAB * AL(1,1,3,I,3,M) * AL(2,1,3,J,3,N)   SS8L564
      IF ( NSTRNG .EQ. 0 )  GO TO 460                                    SS8L565
      DO 460 L=1,NSTRNG                                                  SS8L566
      TT(II,JJ) = TT(II,JJ) + ( AL(1,1,3,I,3,M) * ( AA * AS(L)           SS8L567
     1          * ESW(J,L) * ESW(N,L) + AB1 * YBARS(L) * AS(L) * (        SS8L568
     2          ESW(J,L) * ESDW(N,L) + ESDW(J,L) * ESW(N,L) ) + AB2 * (   SS8L569
     3          XIZZS(L) + XIYYS(L) ) * ESDW(J,L) * ESDW(N,L) ) + A1 *   SS8L570
     4          XIYYS(L) * AL(1,2,3,I,3,M) * ESW(J,L) *ESW(N,L))*RHOS(L)SS8L571
  460 CONTINUE                                                           SS8L572
      IF ( NRING  .EQ. 0 )  GO TO 470                                    SS8L573
      DO 470 K=1,NRING                                                   SS8L574
      TT(II,JJ) = TT(II,JJ) + RHOR(K) * ( AL(2,1,3,J,3,N) * ( BB * AR(K)SS8L575
     1          * ERW(I,K) * ERW(M,K) + XBARR(K) * A1B * AR(K) * (        SS8L576
     2          ERW(I,K) * ERDW(M,K) + ERDW(I,K) * ERW(M,K) )            SS8L577
     2          + A2B * ( XIXXR(K) + XIZZR(K) ) *                        SS8L578
     3          ERDW(I,K) * ERDW(M,K) ) + B1 * AL(2,2,3,J,3,N)*XIXXR(K)  SS8L579
     4          * ERW(I,K) * ERW(M,K) )                                  SS8L580
  470 CONTINUE                                                           SS8L581
      IF ( NLMASS .EQ. 0 )  GO TO 480                                    SS8L582
      DO 480 L=1,NLMASS                                                  SS8L583
      TT(II,JJ) = TT(II,JJ) + PMASS(L) * EVAL(1,1,3,I,IPWW(L)) *         SS8L584
     1EVAL(1,2,3,J,IPWY(L))*EVAL(1,1,3,M,IPWW(L))*EVAL(1,2,3,N,IPWY(L))  SS8L585
  480 CONTINUE                                                           SS8L586
      IF ( IEDGE  .EQ. 0 )  GO TO 510                                    SS8L587
      X(45) = 0.                                                         SS8L588
      DO 490 L=1,NPNX                                                    SS8L589
      DO 490 K=1,NPNY                                                    SS8L590
      X(45) = X(45) - PX(L,K) * $W(L,1,2,I,M) * $W(K,2,1,J,N) * A1B      SS8L591
     1               - PY(L,K) * $W(L,1,1,I,M) * $W(K,2,2,J,N) * AB1     SS8L592
     2               -PXY(L,K) *($W(L,1,3,I,M) * $W(K,2,3,N,J)           SS8L593
     3                         + $W(L,1,3,M,I) * $W(K,2,3,J,N) )          SS8L594
  490 CONTINUE                                                           SS8L595
      U(KK,LL) = X(45)                                                   SS8L596
      IF ( NSTRNG .EQ. 0 )  GO TO 500                                    SS8L597
      DO 500 L=1,NSTRNG                                                  SS8L598
      U(KK,LL) = U(KK,LL) - PAXS(L) * AL(1,2,3,I,3,M) * A1               SS8L599
     1          * ESW(J,L) * ESW(N,L)                                    SS8L600
  500 CONTINUE                                                           SS8L601
      IF ( NRING  .EQ. 0 )  GO TO 510                                    SS8L602
      DO 510 K=1,NRING                                                   SS8L603
      U(KK,LL) = U(KK,LL) - PAXR(K) * B1 * AL(2,2,3,J,3,N)               SS8L604
     1          * ERW(I,K) * ERW(M,K)                                    SS8L605
  510 CONTINUE                                                           SS8L606
      IF ( IFLAGW .EQ. 0 )  GO TO 1000                                   SS8L607
      IF ( JJ .GT. NTUX*NTUY + NTVX*NTVY + 1 )  GO TO 1000               SS8L608
      IF ( IFLAGW .EQ. 2 )  GO TO 521                                    SS8L609
      X(46) = 0.                                                         SS8L610
      DO 520 K=1,NQTX                                                    SS8L611
      DO 520 L=1,NQTY                                                    SS8L612
  520 X(46) =  X(46) + QQ(K,L) * AB * P(K,1,1,3,I) * P(L,2,1,3,J)        SS8L613
      Q(II) = X(46)                                                      SS8L614
  521 CONTINUE                                                           SS8L615
```

```
      IF ( IEDGE  .EQ. 0 )  GO TO 525
      IF ( NSTRNG .EQ. 0 )  GO TO 522
      DO 522 L=1,NSTRNG
      S(II) = S(II) + PAXS(L) * A1 * ZBARS(L) * P(1,1,3,3,I) * ESW(J,L)
  522 CONTINUE
      IF ( NRING .EQ. 0 )  GO TO 523
      DO 523 K=1,NRING
      S(II) = S(II) - PAXR(K) * ( - ZBARR(K) * P(1,2,3,3,J)
     1               * ERW(I,K)        + P(1,2,1,3,J) * ( BR1
     2               * ERW(I,K) + BR1 * XBARR(K) * ERDW(I,K) ) )
  523 CONTINUE
      DO 524 K=1,NPNX
      DO 524 L=1,NPNY
  524 S(II) = S(II) - ABR1 * PY(K,L) * P(K,1,1,3,I) * P(L,2,1,3,J)
  525 IF ( NPTLDS .EQ. 0 )  GO TO 530
      DO 530 L=1,NPTLDS
      Q(II) = Q(II) + PC(L) * EVAL(1,1,3,I,IPXX(L))
     1       * EVAL(1,2,3,J,IPYY(L))
  530 CONTINUE
      IF ( NPTMOM .EQ. 0 )  GO TO 550
      DO 550 L=1,NPTMOM
      IF ( ITAGCM(L) .EQ. 1 )  GO TO 540
C     TAG = 1  FOR MY , = 2  FOR MX
      Q(II) = Q(II) - A1 * FC(L)
     1       * EVAL(2,1,3,I,IFXX(L)) * EVAL(1,2,3,J,IFYY(L))
      GO TO 550
  540 Q(II) = Q(II) - B1 * FC(L)
     1       * EVAL(1,1,3,I,IFXX(L)) * EVAL(2,2,3,J,IFYY(L))
  550 CONTINUE
      IF ( NLNMOM .EQ. 0 )  GO TO 1000
      DO 570 L=1,NLNMOM
      IF ( ITAGLM(L) .EQ. 1 )  GO TO 560
      Q(II) = Q(II) - A1B * PLMOM(L) * P(1,2,1,3,J)
     1               * EVAL(2,1,3,I,IDISLM(L))
      GO TO 570
  560 Q(II) = Q(II) - AB1 * PLMOM(L) * P(1,1,1,3,I)
     1               * EVAL(2,2,3,J,IDISLM(L))
  570 CONTINUE
      GO TO 1000
  580  V(II,JJ) =  V(JJ,II)
      IF ( IFLAGD .EQ. 0 )  GO TO 1000
      TT(II,JJ) = TT(JJ,II)
 1000 CONTINUE
      CALL STATUS ( ITIME )
      TIME(6) = .01*ITIME(8)
      ET = TIME(6) - TIME(5)
C **  CHANGE SIGN ON Q
      DO 1584 I=1,MATSIZ
 1584 Q(I) = -Q(I)
      DO 2584 I=1,MWSIZ
      DO 2584 J=1,MWSIZ
 2584 U(I,J) = -U(I,J)
      DO 585 I=1,MATSIZ
      QHOLD(I) = Q(I)
      SHOLD(I) = S(I)
      DO 585 J=1,MATSIZ
```

```
  585 VHOLD(I,J) = V(I,J)
C
      IF ( INTPRT .NE. 1 )  GO TO 670
      WRITE (6,590) ET
  590 FORMAT ('OTIME REQUIRED TO ASSEMBLE MATRICES = ',F7.3,' SEC.')
      WRITE (6,610)  NAMEV
  610 FORMAT ('1MATRIX ',A4)
      DO 630 I=1,MATSIZ
      WRITE (6,620)
  620 FORMAT ('0')
  630 WRITE (6,640) ( V(I,J), J=1,MATSIZ )
  640 FORMAT (' ',10E12.4)
      IF ( IFLAGD .EQ. 0 )  GO TO 651
      WRITE (6,610)  NAMETT
      DO 650 I=1,MATSIZ
      WRITE (6,620)
  650 WRITE (6,640) ( TT(I,J), J=1,MATSIZ)
  651 CONTINUE
      IF ( IEDGE  .EQ. 0 )  GO TO 661
      WRITE (6,610)  NAMEU
      DO 660 I=1,MWSIZ
      WRITE (6,620)
  660 WRITE (6,640) ( U(I,J), J=1,MWSIZ )
  661 CONTINUE
      IF ( IFLAGW .EQ. 0 )  GO TO 670
      WRITE (6,610)  NAMES
      WRITE (6,640)  ( S(J), J=1,MATSIZ )
      WRITE (6,610)  NAMEQ
      WRITE (6,640) ( Q(J), J=1,MATSIZ )
  670 CONTINUE
      RETURN
      END
```

CC = 00704

```
      SUBROUTINE SOLVE
C
      DIMENSION    V(150,150), T(150,150), Z(150,150)
      DIMENSION    VV(22500), TV(22500), ZV(22500)
      DIMENSION    Z1(100,100),Z2(100,50),Z3(50,100),Z4(50,50)
      DIMENSION U(50,50), Q(150)
      DIMENSION    WORK1(150),     WORK2(150)
      DIMENSION    S(150)
      DIMENSION    ITIME(12),      TIME(50)
      DIMENSION    INDEX(150)
C
      COMMON              U
      COMMON / BLOCK  / T
      COMMON / ARRAYS / V
      COMMON / CNTROL / IFLAGD, IFLAGB, IFLAGW, IBCX, IBCY, I$, IEDGE,
     1             J$(2), KEY, K$(2), INTPRT, IKDF, IFLEX
      COMMON / NUMBER / ND2(6), NTWY, NMODES, ND3(12), NUVW, NUV, NW
      COMMON / NUMBER / ITX, ITY
      COMMON / ZWORK  / Z
      COMMON / PARAM  / Q,           S,          WORK1,     WORK2
      COMMON / $TIME  / TIME,      ITIME
      COMMON / MODES / MM(50), NN(50)
C
      EQUIVALENCE  ( Z1(1), Z(1) ), ( Z2(1), Z(10001) )
      EQUIVALENCE  ( Z3(1), Z(15001) ), ( Z4(1), Z(20001) )
      EQUIVALENCE  ( V(1),VV(1) ), ( T(1),TV(1) ), ( Z(1),ZV(1) )
C
      CALL STATUS ( ITIME )
      TIME(10) = .01*ITIME(8) - TIME(1)
      IF ( INTPRT .EQ. 1 )  WRITE (6,10)  TIME(10)
   10 FORMAT ('OELAPSED TIME AT BEGINNING OF ',7H'SOLVE',' = ',F7.2)
C
      IF ( IFLAGW .NE. 0 )  GO TO 20
      IF ( IFLAGD .NE. 0 )  GO TO 90
      IF ( IFLAGB .NE. 0 )  GO TO 170
C
C **  STATIC DEFLECTION
C
   20 CONTINUE
      IF ( IEDGE .EQ. 1 )  GO TO 40
      DO 30 I=1,NUVW
      DO 30 J=1,NUVW
   30 T(I,J) = V(I,J)
      GO TO 65
   40 DO 60 I=1,NUVW
      DO 60 J=1,NUVW
      IF ( I.GT.NUV .AND. J.GT.NUV )  GO TO 50
      T(I,J) = V(I,J)
      GO TO 60
   50 K = I-NUV
      L = J-NUV
      T(I,J) = V(I,J) + U(K,L)
   60 CONTINUE
   65 CONTINUE
C
      IF ( IFLEX.EQ. 0 )  GO TO 70
```

```
      CALL REDUCE ( 1,V,Z1,Z2,Z3,Z4,WORK1,WORK2,NUV,NW )
      CALL FLEX
   70 CONTINUE
C
      DO 80 I=1,NUVW
   80 WORK1(I) = -S(I) - Q(I)
C
      CALL SWITCH ( T, NUVW, 150, 0., 1. )
      CALL SIMEQ ( T,WORK1,NUVW,1,150,150,0.,IER )
      KEY = 1
      GO TO 1000
C
C **  VIBRATION
C
   90 CONTINUE
      CALL STATUS ( ITIME )
      TIME(11) = .01*ITIME(8) - TIME(1)
      DO 100 I=1,NUVW
      DO 100 J=1,NUVW
  100 Z(I,J) = V(I,J)
      CALL SWITCH ( T, NUVW, 150, 0., 1. )
C
      CALL ARRAY (2,NUVW,NUVW,150,150,VV,V)
      CALL ARRAY (2,NUVW,NUVW,150,150,ZV,Z)
      CALL ARRAY (2,NUVW,NUVW,150,150,TV,T)
      CALL NROOT (NUVW,ZV,TV,WORK1,VV)
      CALL ARRAY (1,NUVW,NUVW,150,150,VV,V)
      DO 120 J=1,NUVW
      WORK2(J) = 1.E+40
      DO 110 I=1,NUVW
      IF ( WORK1(I).GE.WORK2(J) )  GO TO 110
      WORK2(J) = WORK1(I)
      INDEX(J) = I
  110 CONTINUE
  120 WORK1(INDEX(J)) = 1.E+40
      DO 130 J=1,NUVW
      WORK1(J) = WORK2(J)
      DO 130 K=1,NUVW
  130 T(J,K) = V(K,INDEX(J))
      CALL STATUS ( ITIME )
      TIME(12) = .01*ITIME(8) - TIME(1)
      ET = TIME(12) - TIME(11)
      IF ( INTPRT .EQ. 1 ) WRITE (6,140) ET
  140 FORMAT ('OTIME TO SOLVE FOR EIGENVALUES AND EIGENVECTORS = ',F7.2)
      DO 55 I=1,NUVW
      BIG = ABS(T(I,NUV+1))
      NSAVE = 1
      DO 59 J=2,NW
      IF (ABS (T(I,J+NUV)).LE.BIG )  GO TO 59
      BIG = ABS ( T(I,J+NUV) )
      NSAVE = J
   59 CONTINUE
      M = ITX
      N = ITY
      IF ( NSAVE .EQ. 1 )  GO TO 3
      DO 2 J=2,NSAVE
```

```
      IF ( N+1-ITY .GE. NTWY )  GO TO 1
      N = N+1
      GO TO 2
    1 N = ITY
      M = M+1
    2 CONTINUE
    3 CONTINUE
      MM(I) = M
      NN(I) = N
      IF ( WORK1(I) .GT. 0. )  WORK1(I)=SQRT(WORK1(I))/6.2831853
   55 CONTINUE
      WRITE (6,160) ( WORK1(I), MM(I), NN(I) , I=1,NUVW )
  160 FORMAT ('1  FREQUENCY',7X,'M',5X,'N'/('0',E13.5,4X,I2,4X,I2))
      KEY = 2
      GO TO 1000
C
C **  BUCKLING
C
  170 CONTINUE
      DO 180 I=1,NW
      DO 180 J=1,NW
  180 U(I,J) = - U(I,J)
C
      IF ( IFLEX .EQ. 0 )  GO TO 190
      CALL REDUCE ( 1,V,Z1,Z2,Z3,Z4,WORK1,WORK2,NUV,NW )
      CALL FLEX
  190 CONTINUE
      IF ( IKDF .NE. 0 )  CALL KDF ( BUCKNX )
C
      IF ( IFLEX .EQ. 0 )  GO TO 200
      CALL YOSFEM ( 2,Z,NW,NW,150,U,NW,50,V,WORK1 )
      GO TO 210
  200 CALL REDUCE ( 2,V,Z1,Z2,Z3,Z4,WORK1,WORK2,NUV,NW )
      CALL YOSFEM ( 2,V,NW,NW,150,U,NW,50,Z,WORK1 )
  210 CONTINUE
      IF ( IFLAGB .EQ. 1 ) CALL EIGONE ( U,WORK1,NW,50 )
      IF ( IFLAGB .EQ. 2 ) CALL EIGALL ( U,WORK1,NW,50,1,2 )
      KEY = 3
 1000 CONTINUE
      RETURN
      END
      SUBROUTINE SWITCH ( DIAG, N, NMAX, FROM, TO )
C     CHANGES A DIAGONAL TERM FROM 0 TO 1 OR FROM 1 TO 0 .
      DIMENSION DIAG(NMAX,N)
      DO 10 I=1,N
      IF ( DIAG(I,I) .EQ. FROM )  DIAG(I,I) = TO
   10 CONTINUE
      RETURN
      END
```

CC = 00161

```
      SUBROUTINE  YOSFEM ( NOPT, A,NRA,NCA,MRA,B,NCB,MRB,C,WORK )       SS8N000
C                                                                       SS8N001
C **  YOSFEM = YE OLDE SUBROUTINE FOR EFFICIENT MULTIPLICATION.         SS8N002
C **  NOPT   = 1, 2, OR 3                                               SS8N003
C **         = 1 , COMPUTES  A = A * B                                  SS8N004
C **         = 2 , COMPUTES  B = A * B                                  SS8N005
C **         = 3 , COMPUTES  C = A * B                                  SS8N006
C **  A      = AN NRA BY NCA MATRIX                                     SS8N007
C **  NRA    = NUMBER OF ROWS IN A                                      SS8N008
C **  NCA    = NUMBER OF COLUMNS IN A                                   SS8N009
C **  MRA    = MAXIMUM NUMBER OF ROWS IN A                              SS8N010
C **  B      = AN NCA BY NCB MATRIX                                     SS8N011
C **  NCB    = NUMBER OF COLUMNS IN B                                   SS8N012
C **  MRB    = MAXIMUM NUMBER OF ROWS IN B                              SS8N013
C **  C      = AN NRA BY NCB MATRIX                                     SS8N014
C **  WORK   = A WORK VECTOR OF LENGTH NRA                              SS8N015
C                                                                       SS8N016
      DIMENSION    A(MRA,NCA), B(MRB,NCB), C(MRA,NCB), WORK(NRA)        SS8N017
C                                                                       SS8N018
      IF ( NOPT .NE. 1 )  GO TO 40                                      SS8N019
      DO 30 I=1,NRA                                                     SS8N020
      DO 20 M=1,NCA                                                     SS8N021
   20 WORK(M) = A(I,M)                                                  SS8N022
      DO 30 J=1,NCB                                                     SS8N023
      A(I,J) = 0.                                                       SS8N024
      DO 30 K=1,NCA                                                     SS8N025
   30 A(I,J) = A(I,J) + WORK(K) * B(K,J)                                SS8N026
      GO TO 100                                                         SS8N027
   40 IF ( NOPT .NE. 2 ) GO TO 70                                       SS8N028
      DO 60 J=1,NCB                                                     SS8N029
      DO 50 M=1,NCA                                                     SS8N030
   50 WORK(M) = B(M,J)                                                  SS8N031
      DO 60 I=1,NRA                                                     SS8N032
      B(I,J) = 0.                                                       SS8N033
      DO 60 K=1,NCA                                                     SS8N034
   60 B(I,J) = B(I,J) + A(I,K) * WORK(K)                                SS8N035
      GO TO 100                                                         SS8N036
   70 DO 80 I=1,NRA                                                     SS8N037
      DO 80 J=1,NCB                                                     SS8N038
      C(I,J) = 0.                                                       SS8N039
      DO 80 K=1,NCA                                                     SS8N040
   80 C(I,J) = C(I,J) + A(I,K) * B(K,J)                                 SS8N041
  100 RETURN                                                            SS8N042
      END                                                               SS8N043
```

CC = 00044

```
      SUBROUTINE  EIGONE  ( A, X, N, NRA )
C
C     THIS SUBROUTINE COMPUTES THE INVERSE OF THE LARGEST EIGEN VALUE
C     OF AN N BY N MATRIX, AND THE CORRESPONDING MODE SHAPE, BY SIMPLE
C     ITERATION.
C     CAST PROBLEM IN THE FORM A*X = X/OLAMB
C
      DIMENSION     A(NRA,N),       X(N)
      DIMENSION     B(150,150),     ITIME(12),      TIME(50)
      DIMENSION     USED(150),      XA(150),        XX(150)
      DIMENSION     XXX(150),       XY(150),        MPN(150)
C
      COMMON / ZWORK  / B
      COMMON / PARAM  / XA,  XX,  USED,  XXX,  XY,  MPN
      COMMON / $TIME  / TIME,     ITIME
      COMMON / CNTROL / I$(12),   INTPRT
C
      CALL STATUS (ITIME)
      TIME(20) = .01*ITIME(8) - TIME(1)
      PDIDLE=.00001
      MAD= 72
      IKEP=1
      OLAMB = 0.
      DO 1 I=1,N
    1 X(I)=.1
      M=1
    6 XMIN=0
      OLAMBO=OLAMB
C     A NEW MODE SHAPE IS COMPUTED AS A TIMES X, AND THE LARGEST ELEMENT
C     OF THE NEW X IS STORED IN XMIN.
      DO 44 I=1,N
   44 XA(I)=X(I)
      DO 42 K=1,6
      DO 3 I=1,N
      XX(I)=0.
      DO 3 J=1,N
    3 XX(I)=XX(I)+ A(I,J)*X(J)
      XPQ=X(N)/XX(N)
      XPR=XPQ/ABS(XPQ)
      DO 41 I=1,N
      XXX(I)=X(I)
   41 X(I)=XX(I)
   42 CONTINUE
      DO 2 I=1,N
      IF(ABS(XMIN)-ABS(XX(I)))7,2,2
    7 XMIN = XX(I)
      JJ= I
      MPN(IKEP)=I
    2 CONTINUE
C     THE NEW VECTOR IS NORMALIZED WITH RESPECT TO XMIN.
      DO 4 I=1,N
    4 XX(I)= XX(I)/ XMIN
C     THE LATEST APPROXIMATION TO 1 DIVIDED BY THE LARGEST EIGEN VALUE
C     IS COMPUTED.
      OLAMB=XA(JJ)/XMIN
      OLAMB=   ((ABS(OLAMB))**.1666667)*XPR
```

```
C      THE NEW VECTOR IS STORED FOR A NEW ITERATION.                    SS8N101
       DO 9 I =1,N                                                      SS8N102
     9 X(I) = XX(I)                                                     SS8N103
C      THE RELATIVE CHANGE IN OLAMB IS THE BASIS FOR CONVERGENCE.       SS8N104
       IF(ABS((OLAMB - OLAMBO) /OLAMB ) .LT.PDIDLE)GO TO 5              SS8N105
       M=M+1                                                            SS8N106
       IF(M.GT.15)PDIDLE =.0005                                         SS8N107
       IF (M.LT.50) GO TO 6                                             SS8N108
       WRITE(6,8)OLAMBO,OLAMB                                           SS8N109
     8 FORMAT( 'ONO CONVERGENCE'2E15.7)                                 SS8N110
       XY(IKEP)=OLAMB                                                   SS8N111
       DO 60 IJ=1,N                                                     SS8N112
    60 B(IKEP,IJ) =X(IJ)                                                SS8N113
       GO TO 39                                                         SS8N114
     5 IF(M.GT.15)GO TO 20                                              SS8N115
       M=M+1                                                            SS8N116
       GO TO 6                                                          SS8N117
    20 IF ( INTPRT .EQ. 1 ) WRITE (6,12) M                              SS8N118
    12 FORMAT( '0'I4,' ITERATIONS')                                     SS8N119
       DO 43 I=1,N                                                      SS8N120
    43 X(I)=(X(I)+XXX(I)/OLAMB/XMIN)/2.                                 SS8N121
       DO 55 I=1,N                                                      SS8N122
    55 B(IKEP,I)=X(I)                                                   SS8N123
       XY(IKEP)=OLAMB                                                   SS8N124
    39 CONTINUE                                                         SS8N125
   500 DO 38 J=1,IKEP                                                   SS8N126
       X(J)=XY(J)                                                       SS8N127
       DO 38 I=1,N                                                      SS8N128
    38 A(J,I)=B(J,I)                                                    SS8N129
    40 CONTINUE                                                         SS8N130
       CALL STATUS (ITIME)                                              SS8N131
       TIME(21) = .01*ITIME(8) - TIME(1)                                SS8N132
       ET = TIME(21) - TIME(20)                                         SS8N133
       IF ( INTPRT .EQ. 1 )  WRITE (6,600)  ET                          SS8N134
   600 FORMAT ('OTIME REQUIRED TO FIND ONE EIGENVALUE AND EIGENVECTOR = 'SS8N135
      1        ,F7.2)                                                   SS8N136
       RETURN                                                           SS8N137
       END                                                              SS8N138

                                                        CC = 00094
```

```
      SUBROUTINE  EIGALL   ( A, X, N, NRA, ITAG, MODES )
C
C **  THIS SUBROUTINE FINDS ALL THE EIGENVALUES OF THE NRA BY N
C **  MATRIX A. IT ALSO FINDS THE EIGENVECTORS CORRESPONDING TO
C **  THE FIRST 'MODES' EIGENVALUES.
C **  THE MATRIX EQUATION IS IN THE FORM  A*X = X/EGNVAL
C
      DIMENSION    A(NRA,N), X(N)
      DIMENSION    USED(150),      XX(150),        WORK(3000)
      DIMENSION    Z(150,150),     XY(150),  NDUM1(150),  NDUM2(150)
      DIMENSION    VEC(150),  ITIME(12),  TIME(50)
C
      COMMON / ZWORK  / Z
      COMMON / PARAM  / XX,  XY,  USED,  VEC,  NDUM1,  NDUM2
      COMMON / EIGWRK / WORK
      COMMON / CNTROL / IFLAGD,   I$(11),   INTPRT
      COMMON / COMMON / DUMCOM(150)
      COMMON / $TIME  / TIME,      ITIME
C
      DO 10 J=1,N
   10 X(J) = 0.
      DO 20 I=1,N
      DO 20 J=1,N
   20 Z(I,J) = A(I,J)
      IPRNT = INTPRT
      CALL STATUS (ITIME)
      TIME(17) = .01*ITIME(8) - TIME(1)
      CALL HESSEN ( Z, N, 150 )
      CALL QREIG  ( Z, N, XY, XX, IPRNT, 150 )
      CALL STATUS ( ITIME )
      TIME(18) = .01*ITIME(8) - TIME(1)
      ET = TIME(18) - TIME(17)
      IF ( INTPRT .EQ. 1 )  WRITE (6,21)  ET
   21 FORMAT ('OTIME REQUIRED TO FIND ALL EIGENVALUES = ',F7.2)
      IF ( ITAG .EQ. 3 )  GO TO 70
      GREAT=  0.
      DO 71 I=1,N
      IF(XX(I).NE.0.)GO TO 71
      IF ( XY(I) .EQ. 0. )  GO TO 71
      IF(ABS(GREAT).GT.ABS(XY(I)))GO TO 71
      GREAT = XY(I)
   71 CONTINUE
      GREAT2 = -0.
      DO 72 I=1,N
      IF(XX(I).NE.0.)GO TO 72
      IF ( XY(I) .EQ. 0. )  GO TO 72
      IF(GREAT*XY(I).GT.0..OR.ABS(GREAT2).GT.ABS(XY(I))) GO TO 72
      GREAT2 = XY(I)
   72 CONTINUE
      MODES = 2
      XY(1)= GREAT
      XY(2)= GREAT2
      X(1)= 1./GREAT
      IF(ABS(GREAT2).LT.1.E-40)GO TO 80
      X(2)= 1./GREAT2
      GO TO 73
```

```
80 X(2)=0.
   MODES = 1
   GO TO 73
70 CONTINUE
   DO 50 J=1,N
   IF ( XY(J) .NE. 0. )  XY(J) = 1./XY(J)
50 CONTINUE
   DO 74 J=1,N
   IF(XX(J).NE.0.)XY(J)= 0.
   DO 75 I=J,N
   IF(XX(I).NE.0.)GO TO 75
   IF ( XY(I) .EQ. 0. )  GO TO 75
   IF(XY(I).LT.XY(J))GO TO 75
   GREAT= XY(J)
   XY(J)= XY(I)
   XY(I)= GREAT
   GREAT= XX(J)
   XX(J)= XX(I)
   XX(I)= GREAT
75 CONTINUE
   IF ( XY(J) .NE. 0. )   X(J) = 1./XY(J)
74 CONTINUE
73 CONTINUE
   DO 77 I=1,MODES
   DO 78 J=1,N
   DO 78 K=1,N
78 Z(J,K) = A(J,K)
   EGNVAL = X(I)
   IF ( ITAG .NE. 3 )  EGNVAL = 1./X(I)
   CALL EGNVCT ( Z, XX, EGNVAL, VEC, NDUM1, NDUM2, N, 150, IPRNT )
   J0 = (I-1)*N
   DO 79 J=1,N
   K = J0 + J
79 WORK(K) = VEC(J)
77 CONTINUE
   DO 90 I=1,MODES
   J = (I-1)*N
   DO 90 K=1,N
   L = J + K
90 A(I,K) = WORK(L)
   CALL STATUS (ITIME)
   TIME(19) = .01*ITIME(8) - TIME(1)
   ET = TIME(19) - TIME(18)
   IF ( INTPRT .EQ. 1 )  WRITE (6,91)  ET
91 FORMAT ('0TIME REQUIRED TO FIND EIGENVECTORS = ',F7.2)
   RETURN
   END
```

CC = 00103

```
      SUBROUTINE EGNVCT ( C1, C2, EIGEN, C3, N1, N2, N, NROWS, NTIME )
C
C     SUBROUTINE TO OBTAIN EIGENVECTOR FROM REAL NON-SYMMETRIC
C     MATRICES FOR WHICH THE EIGENVALUE IS KNOWN.  THE METHOD
C     USED IS THE DIRECT METHOD OUTLINED IN ERR-FW-    BY DR.
C     A. M. CUNNINGHAM. ALL ARITHMETIC IS IN DOUBLE PRECISION.
C
      DIMENSION C1(NROWS,NROWS), C2(NROWS), C3(NROWS), N1(NROWS),
     1          N2(NROWS)
C
      II3 = N
      II2 = N - 1
      IF (NTIME .NE. 0 ) CALL STATUS (N1)
      IT1 = N1(8)
      D1 = 0.0 D0
      DO 20 J=1,N
      N1(J) = J
      N2(J) = J
      C1(J,J) = C1(J,J) - EIGEN
      DO 10 I=1,N
      D2 =  ABS(C1(I,J))
      IF (D1-D2) 5,10,10
    5 D1 = D2
      I1 = I
      J1 = J
   10 CONTINUE
   20 CONTINUE
      DO 150 K6=2,N
      IF (C1(I1,J1)) 50,30,50
   30 K5 = K6 - 1
   35 WRITE (6,40) K5
   40 FORMAT (1H1, 4X,57H THE REDUCED MATRIX WAS FOUND TO BE SINGULAR ON
     1 ITERATION, I4 )
      N1(1) = 1
      GO TO 1000
C
   50 D1 = 1.0/C1(I1,J1)
      D2 = C1(I1,II3)
      D3 = C1(II3,J1)
      D4 = C1(II3,II3)
      DO 60 I=1,II2
      C3(I)     = C1(I,J1)
      C1(I,J1) = C1(I,II3)
      C1(I,II3)  = -C3(I)*D1
      D5         = -C1(I1,I)*D1
      C1(I1,I) = C1(II3,I)
      C1(II3,I) = D5
   60 CONTINUE
      C3(I1)       = D3
      C1(I1,J1)   = D4
      C1(II3,J1) = -D2*D1
      C1(I1,II3) = -D3*D1
      C1(II3,II3)=  D1
      IF (II3 .EQ. N) GO TO 80
      II4 = II3 + 1
      DO 70 I=II4,N
```

```
      D6 = C1(I1,I)                                                     SS8Q056
      C1(I1,I)  = C1(II3,I)                                             SS8Q057
      C1(II3,I) = D6                                                    SS8Q058
      C3(I)     = C1(I,J1)                                              SS8Q059
      C1(I,J1)  = C1(I,II3)                                             SS8Q060
   70 C1(I,II3) = C3(I)                                                 SS8Q061
   80 I = N1(J1)                                                        SS8Q062
      N1(J1) = N1(II3)                                                  SS8Q063
      N1(II3) = I                                                       SS8Q064
      I = N2(I1)                                                        SS8Q065
      N2(I1) = N2(II3)                                                  SS8Q066
      N2(II3) = I                                                       SS8Q067
      D7 = 0.0 D0                                                       SS8Q068
      DO 140 J=1,II2                                                    SS8Q069
      D8 = C1(II3,J)                                                    SS8Q070
      DO 130 I=1,II2                                                    SS8Q071
      C1(I,J) = C1(I,J) + C3(I)*D8                                      SS8Q072
      D9 =  ABS(C1(I,J))                                                SS8Q073
      IF (D7-D9) 120,130,130                                            SS8Q074
  120 D7 = D9                                                           SS8Q075
      I1 = I                                                            SS8Q076
      J1 = J                                                            SS8Q077
  130 CONTINUE                                                          SS8Q078
  140 CONTINUE                                                          SS8Q079
      II3 = II3 - 1                                                     SS8Q080
      II2 = II2 - 1                                                     SS8Q081
  150 CONTINUE                                                          SS8Q082
C                                                                       SS8Q083
C                                                                       SS8Q084
  160 C3(2) = C1(2,1)                                                   SS8Q085
      C3(1) = 1.0                                                       SS8Q086
      DO 180 J=3,N                                                      SS8Q087
      C3(J) = 0.0 D0                                                    SS8Q088
      J1 = J-1                                                          SS8Q089
      DO 170 I=1,J1                                                     SS8Q090
      C3(J) = C3(J) + C3(I)*C1(J,I)                                     SS8Q091
  170 CONTINUE                                                          SS8Q092
  180 CONTINUE                                                          SS8Q093
      IF ( ABS(C1(1,1) ) .LT. 1.0 E-20 ) GO TO 202                      SS8Q094
      DO 201 K6=1,2                                                     SS8Q095
C                                                                       SS8Q096
      DO 184 J=1,N                                                      SS8Q097
      I1 = N2(J)                                                        SS8Q098
      DO 182 I=1,N                                                      SS8Q099
      IF ( I1 .EQ. N1(I) ) GO TO 184                                    SS8Q100
  182 CONTINUE                                                          SS8Q101
  184 C2(J) = C3(I)                                                     SS8Q102
C                                                                       SS8Q103
      DO 190 J=2,N                                                      SS8Q104
      I1 = N - J + 1                                                    SS8Q105
      J1 = I1 + 1                                                       SS8Q106
      DO 185 I=1,I1                                                     SS8Q107
      C2(I) = C2(I) + C1(I,J1)*C2(J1)                                   SS8Q108
  185 CONTINUE                                                          SS8Q109
  190 CONTINUE                                                          SS8Q110
      D1  =  C1(1,1)/C2(1)                                              SS8Q111
```

```
      C3(1) = 1.0 D0                                                    SS8Q112
      DO 200 J=2,N                                                      SS8Q113
      I1 = J - 1                                                        SS8Q114
      C3(J) = C2(J)*C1(J,J)*D1                                          SS8Q115
      DO 195 I=1,I1                                                     SS8Q116
      C3(J) = C3(J) + C1(J,I)*C3(I)                                     SS8Q117
  195 CONTINUE                                                          SS8Q118
  200 CONTINUE                                                          SS8Q119
  201 CONTINUE                                                          SS8Q120
C                                                                       SS8Q121
C     C3(I) NOW CONTAINS THE EIGENVECTOR WHICH MUST BE RE-ARRANGED      SS8Q122
C     ACCORDING TO THE ORDER DICTATED BY N1(I) BACK TO THE ORIGINAL     SS8Q123
C     ORDER.                                                            SS8Q124
C                                                                       SS8Q125
  202 DO 230 I=1,N                                                      SS8Q126
      I1 = N1(I)                                                        SS8Q127
      N1(I) = I                                                         SS8Q128
  205 IF (I1-I) 210,230,210                                             SS8Q129
  210 D1     = C3(I1)                                                   SS8Q130
      C3(I1) = C3(I)                                                    SS8Q131
      C3(I)  = D1                                                       SS8Q132
      K = N1(I1)                                                        SS8Q133
      N1(I1) = I1                                                       SS8Q134
      I1 = K                                                            SS8Q135
      GO TO 205                                                         SS8Q136
  230 CONTINUE                                                          SS8Q137
C                                                                       SS8Q138
      IF (NTIME) 240,260,240                                            SS8Q139
  240 CALL STATUS (N1)                                                  SS8Q140
      A1 = (N1(8) - IT1)*0.01                                           SS8Q141
      WRITE (6,250) N,A1                                                SS8Q142
  250 FORMAT ( 1H0,////,42H    THE TOTAL TIME FOR OBTAINING THE         SS8Q143
     1           ,//,   25H    EIGENVECTOR OF ORDER ,I3,6H  IS   ,E12.5,SS8Q144
     2                  9H SECONDS.  )                                  SS8Q145
  260 N1(1) = 2                                                         SS8Q146
C                                                                       SS8Q147
 1000 RETURN                                                            SS8Q148
      END                                                               SS8Q149
```

CC = 00150

```
      SUBROUTINE  DISPLA  ( C, ITAG )
C
C **  THIS SUBROUTINE CALCULATES AND PRINTS DEFLECTIONS, CURVATURES,
C **  MOMENTS, SHEARS AND EDGE REACTIONS
C
      DIMENSION    F(15,25,25),   FMAX(15),      $(3,4,4),      C(150)
      DIMENSION    RA(2,25),      RB(2,25),      RLN(25)
      DIMENSION    A(3,3),        B(3,3),        D(3,3)
      DIMENSION    PKC(50),       IGSPRX(50),    IGSPRY(50)
      DIMENSION    PLINE(50),     IDISLS(50),    ITAGLS(50)
      DIMENSION    E(4,2,3,10,25)
C
      COMMON / ARRAYS / F,        FMAX
      COMMON / VALUES / E
      COMMON / PARAM /  H(2250),  PKC,      IGSPRX,   IGSPRY,
     1                  PLINE,    IDISLS,   ITAGLS
      COMMON / ABD    / A,        B,        D
      COMMON / GEOM   / AA,       BB,       RR
      COMMON / CNTROL / NCNT(7),  IREACT,   IOUT
      COMMON / NUMBER / NPLYS,    NTUX,     NTVX,     NTWX,     NTUY,
     1                  NTVY,     NTWY,     NMODES,   NNUM(10), NPTSUP,
     2                  NLNSPR,   NUVW,     NUV,      NW
C
      EQUIVALENCE  (H(1),RA(1)),(H(51),RB(1)),(H(101),RLN(1))
      EQUIVALENCE  (H(126),$(1))
      DATA  NMW / 'W' /, NMU / 'U' /, NMV / 'V' /
C
      ITHERY = 1
   40 DO 100 K=1,25
      DO 100 L=1,25
      DO 41 K1=1,3
      DO 41 K2=1,4
      DO 41 K3=1,4
   41 $(K1,K2,K3) = 0.
      M = 1
      IF ( ITAG .EQ. 3 )  M = 3
      IF ( IOUT .EQ. 1 .AND. IREACT .EQ. 0 )  M=3
   42 DO 80 N=M,3
      DO 80 I=1,NTWX
      DO 80 J=1,NTWY
      IF ( N.EQ.1 )  II = (I-1)*NTUY + J
      IF ( N.EQ.2 )  II = NTUX*NTUY + (I-1)*NTVY +J
      IF ( N.EQ.3 )  II = NUV + (I-1)*NTWY + J
      IF ( N.EQ.3 .AND. ITAG.EQ.3 )  II = II - NUV
      IF ( N.EQ.1 )  GO TO 50
      IF ( N.EQ.2 )  GO TO 60
      IF ( N.EQ.3 )  GO TO 70
   50 $(N,2,1) = $(N,2,1) + E(2,1,N,I,K)*E(1,2,N,J,L)*C(II)  /AA
      $(N,3,1) = $(N,3,1) + E(3,1,N,I,K)*E(1,2,N,J,L)*C(II)  /AA/AA
      $(N,2,2) = $(N,2,2) + E(2,1,N,I,K)*E(2,2,N,J,L)*C(II)  /AA/BB
      $(N,1,3) = $(N,1,3) + E(1,1,N,I,K)*E(3,2,N,J,L)*C(II)  /BB/BB
      $(N,1,2) = $(N,1,2) + E(1,1,N,I,K)*E(2,2,N,J,L)*C(II)  /BB
      $(N,1,1) = $(N,1,1) + E(1,1,N,I,K)*E(1,2,N,J,L)*C(II)
      GO TO 80
   60 $(N,2,1) = $(N,2,1) + E(2,1,N,I,K)*E(1,2,N,J,L)*C(II)  /AA
      $(N,3,1) = $(N,3,1) + E(3,1,N,I,K)*E(1,2,N,J,L)*C(II)  /AA/AA
```

```
      $(N,2,2) = $(N,2,2) + E(2,1,N,I,K)*E(2,2,N,J,L)*C(II)  /AA/BB
      $(N,1,3) = $(N,1,3) + E(1,1,N,I,K)*E(3,2,N,J,L)*C(II)  /BB/BB
      $(N,1,2) = $(N,1,2) + E(1,1,N,I,K)*E(2,2,N,J,L)*C(II)  /BB
      $(N,1,1) = $(N,1,1) + E(1,1,N,I,K)*E(1,2,N,J,L)*C(II)
      GO TO 80
   70 $(N,1,1) = $(N,1,1) + E(1,1,N,I,K)*E(1,2,N,J,L)*C(II)
      IF ( IOUT .EQ. 1 .AND. IREACT .EQ. 0 )  GO TO 80
      $(N,2,1) = $(N,2,1) + E(2,1,N,I,K)*E(1,2,N,J,L)*C(II)  /AA
      $(N,3,1) = $(N,3,1) + E(3,1,N,I,K)*E(1,2,N,J,L)*C(II)  /AA/AA
      $(N,4,1) = $(N,4,1) + E(4,1,N,I,K)*E(1,2,N,J,L)*C(II)  /AA/AA/AA
      $(N,3,2) = $(N,3,2) + E(3,1,N,I,K)*E(2,2,N,J,L)*C(II)  /AA/AA/BB
      $(N,2,2) = $(N,2,2) + E(2,1,N,I,K)*E(2,2,N,J,L)*C(II)  /AA/BB
      $(N,2,3) = $(N,2,3) + E(2,1,N,I,K)*E(3,2,N,J,L)*C(II)  /AA/BB/BB
      $(N,1,4) = $(N,1,4) + E(1,1,N,I,K)*E(4,2,N,J,L)*C(II)  /BB/BB/BB
      $(N,1,3) = $(N,1,3) + E(1,1,N,I,K)*E(3,2,N,J,L)*C(II)  /BB/BB
      $(N,1,2) = $(N,1,2) + E(1,1,N,I,K)*E(2,2,N,J,L)*C(II)  /BB
   80 CONTINUE
      F(1,K,L) = $(3,1,1)
      IF ( IOUT .EQ. 1 .AND. IREACT .EQ. 0 )  GO TO 100
      F(2,K,L) = $(1,1,1)
      F(3,K,L) = $(2,1,1)
      IF ( IOUT .EQ. 2 .AND. IREACT .EQ. 0 )  GO TO 100
      EX = $(1,2,1)
      EY = $(2,1,2) + $(3,1,1)/RR
      EXY= $(1,1,2) + $(2,2,1)
      XK =-$(3,3,1)
      IF ( ITHERY .NE. 1 ) GO TO 85
      YK = $(2,1,2)/RR - $(3,1,3)
      XYK= 2.*( $(2,2,1)/RR - $(3,2,2) )
      GO TO 86
   85 YK = -$(3,1,3) - $(3,1,1)/RR/RR
      XYK = -2.*$(3,2,2) - $(1,1,2)/RR + $(2,2,1)/RR
   86 CONTINUE
      F(4,K,L) = EX
      F(5,K,L) = EY
      F(6,K,L) = EXY
      F(7,K,L) = XK
      F(8,K,L) = YK
      F(9,K,L) = XYK
      IF ( IOUT .EQ. 3 .AND. IREACT .EQ. 0 )  GO TO 100
   90 F(10,K,L)= B(1,1)*EX+B(1,2)*EY+B(1,3)*EXY+D(1,1)*XK+D(1,2)*YK
     1                                                  +D(1,3)*XYK
      F(11,K,L)= B(1,2)*EX+B(2,2)*EY+B(2,3)*EXY+D(1,2)*XK+D(2,2)*YK
     1                                                  +D(2,3)*XYK
      F(12,K,L)= B(1,3)*EX+B(2,3)*EY+B(3,3)*EXY+D(1,3)*XK+D(2,3)*YK
     1                                                  +D(3,3)*XYK
C **  LET $(1,4,4) = MX,X
C **      $(2,4,4) = MY,Y
C **      $(3,4,4) = MXY,X
C **      $(3,4,3) = MXY,Y
      $(1,4,4) = B(1,1)*$(1,3,1) + B(1,2)*($(2,2,2)+$(3,2,1)/RR)
     1          + B(1,3)*($(1,2,2)+$(2,3,1)) - D(1,1)*$(3,4,1)
     2          + D(1,2)*($(2,2,2)/RR-$(3,2,3)) + D(1,3)*2.*($(2,3,1)/RR
     3          - $(3,3,2) )
      $(2,4,4) = B(1,2)*$(1,2,2) + B(2,2)*($(2,1,3)+$(3,1,2)/RR)
     1          + B(2,3)*($(1,1,3)+$(2,2,2)) - D(1,2)*$(3,3,2)
```

```
     2           + D(2,2)*($(2,1,3)/RR-$(3,1,4))
     3           + D(2,3)*2.*($(2,2,2)/RR - $(3,2,3) )
      $(3,4,4) = B(1,3)*$(1,3,1) + B(2,3)*($(2,2,2)+$(3,2,1)/RR)
     1           + B(3,3)*($(1,2,2)+$(2,3,1)) - D(1,3)*$(3,4,1)
     2           + D(2,3)*($(2,2,2)/RR-$(3,2,3))
     3           + D(3,3)*2.*($(2,3,1)/RR-$(3,3,2))
      $(3,4,3) = B(1,3)*$(1,2,2) + B(2,3)*($(2,1,3)+$(3,1,2)/RR)
     1           + B(3,3)*($(1,1,3)+$(2,2,2)) - D(1,3)*$(3,3,2)
     2           + D(2,3)*($(2,1,3)/RR-$(3,1,4))
     3           + D(3,3)*2.*($(2,2,2)/RR-$(3,2,3))
C
C     F(13,K,L) = QX = MX,X + MXY,Y
C     F(14,K,L) = QY = MY,Y + MXY,X
C
      F(13,K,L) = $(1,4,4) + $(3,4,3)
      F(14,K,L) = $(2,4,4) + $(3,4,4)
C
C     RA = QX + MXY,Y
C     RB = QY + MXY,X
C
      IF(K.EQ.1)  RA(1,L) = - ( F(13,K,L)+ $(3,4,3) )
      IF(K.EQ.25) RA(2,L) =     F(13,K,L)+ $(3,4,3)
      IF(L.EQ.1)  RB(1,K) = - ( F(14,K,L)+ $(3,4,4) )
      IF(L.EQ.25) RB(2,K) =     F(14,K,L)+ $(3,4,4)
  100 CONTINUE
C
C     TO NORMALIZE
C
      KMAX = 14
      IF ( IREACT .NE. 0 )  GO TO 101
      IF ( IOUT .EQ. 1 )  KMAX = 1
      IF ( IOUT .EQ. 2 )  KMAX = 3
      IF ( IOUT .EQ. 3 )  KMAX = 9
  101 CONTINUE
      CALL NRMLIZ ( 1, KMAX )
      WRITE (6,600)  NMW, FMAX(1)
  600 FORMAT ('1THE ',A1,' DEFLECTIONS DIVIDED BY ',E15.6,'/10000 FOLLOW
     1')
      CALL  OUT ( 1)
      I=IOUT
      IF(I.EQ.1.OR.I.EQ.6.OR.I.EQ.7.OR.I.EQ.8)  GO TO 150
      WRITE (6,600)  NMU, FMAX(2)
      CALL  OUT ( 2)
      WRITE (6,600)  NMV, FMAX(3)
      CALL  OUT ( 3)
      IF ( IOUT .EQ. 2 .OR. IOUT .EQ. 3 )  GO TO 150
  220 WRITE (6,680) FMAX(10)
  680 FORMAT ('1MX DIVIDED BY ',E15.6,'/10000 FOLLOWS')
      CALL  OUT (10)
      WRITE (6,690) FMAX(11)
  690 FORMAT ('1MY DIVIDED BY ',E15.6,'/10000 FOLLOWS')
      CALL  OUT (11)
      WRITE (6,700) FMAX(12)
  700 FORMAT ('1MXY DIVIDED BY ',E15.6,'/10000 FOLLOWS')
      CALL  OUT (12)
      WRITE (6,710) FMAX(13)
```

```
  710 FORMAT ('1QX DIVIDED BY ',E15.6,'/10000 FOLLOWS')
      CALL  OUT (13)
      WRITE (6,720) FMAX(14)
  720 FORMAT ('1QY DIVIDED BY ',E15.6,'/10000 FOLLOWS')
      CALL  OUT (14)
  150 IF ( IREACT .EQ. 0 )  GO TO 900
C **  POINT SPRING REACTIONS
      IF ( NPTSUP .EQ. 0 )  GO TO 170
      DO 160 J=1,NPTSUP
      K = IGSPRX(J)
      L = IGSPRY(J)
      FD = - F(1,K,L) * PKC(J) * FMAX(1)
  160 WRITE (6,650) K,L,FD
  650 FORMAT ( 'OTHE REACTION AT GRID POINT 'I3,','I3,' IS ',E15.7)
  170 CONTINUE
      IF ( NLNSPR .EQ. 0 )  GO TO 725
      DO 210 J=1,NLNSPR
      IF ( ITAGLS(J) .EQ. 2 )  GO TO 190
      L = IDISLS(J)
      DO 180 K=1,25
  180 RLN(K) = - F(1,K,L) * PLINE(J) * FMAX(1)
      WRITE (6,660) L, ( RLN(K), K=1,25 )
  660 FORMAT ( 'OTHE REACTION OF THE LINE SPRING ALONG GRID LINE 'I3,
     1        ' PARALLEL TO THE X AXIS FOLLOWS'/(5E15.7))
      GO TO 210
  190 K = IDISLS(J)
      DO 200 L=1,25
  200 RLN(L) = - F(1,K,L) * PLINE(J) * FMAX(1)
      WRITE (6,670) K, ( RLN(L), L=1,25 )
  670 FORMAT ('OTHE REACTION OF THE LINE SPRING ALONG GRID LINE 'I3,
     1        ' PARALLEL TO THE Y AXIS FOLLOWS'/(5E15.7))
  210 CONTINUE
C **  CORNER REACTIONS
  725 F(12,1,1)   = -2. *F(12,1,1)    * FMAX(12)
      F(12,1,25)  =  2. * F(12,1,25)  * FMAX(12)
      F(12,25,1)  =  2. * F(12,25,1)  * FMAX(12)
      F(12,25,25) = -2. * F(12,25,25) * FMAX(12)
      WRITE (6,730)  ( RA(1,L), L=1,25 )
  730 FORMAT(1H1'THE REACTIONS ALONG X=0 FOLLOW'/(1H07E16.7))
      WRITE (6,740)  ( RA(2,L), L=1,25 )
  740 FORMAT(1H0/' THE REACTIONS ALONG X=A FOLLOW'/(1H07E16.7))
      WRITE (6,750)  ( RB(1,K), K=1,25 )
  750 FORMAT(1H0/' THE REACTIONS ALONG Y=0 FOLLOW'/(1H07E16.7))
      WRITE (6,760)  ( RB(2,K), K=1,25 )
  760 FORMAT(1H0/' THE REACTIONS ALONG Y=B FOLLOW'/(1H07E16.7))
      WRITE (6,770) F(12,1,1)
  770 FORMAT(1H0/' THE CORNER REACTION AT 0,0 IS' E16.7)
      WRITE (6,780) F(12,25,1)
  780 FORMAT(1H /' THE CORNER REACTION AT A,0 IS' E16.7)
      WRITE (6,790) F(12,1,25)
  790 FORMAT(1H /' THE CORNER REACTION AT 0,B IS' E16.7)
      WRITE (6,800) F(12,25,25)
  800 FORMAT(1H /' THE CORNER REACTION AT A,B IS' E16.7)
  900 IF ( IOUT .GE. 3 )  CALL STRESS
  999 RETURN
      END
```

CC = 00224

```
      SUBROUTINE PRINT
C **
C **  THIS SUBROUTINE CONTROLS THE PRINTING OF GRID POINT OUTPUT.
C **
      DIMENSION T(150,150)
      DIMENSION  U(50,50), Q(150), S(150)
      DIMENSION    WORK1(150),    WORK2(150),    ITIME(12),    TIME(50)
      DOUBLE PRECISION T
C
      COMMON              U
      COMMON / BLOCK  / T
      COMMON / CNTROL / ID, IFLAGB, I$(7), KEY
      COMMON / NUMBER / N$(6),NTWY,NMODES,M$(12),NUVW,NUV,NW,ITX,ITY
      COMMON / $TIME  / TIME, ITIME
      COMMON / PARAM  / Q,         S,         WORK1,    WORK2
      COMMON / MODES / MM(50), NN(50)
C
      IF ( KEY .EQ. 1 )  GO TO 10
      IF ( KEY .EQ. 2 )  GO TO 20
      IF ( KEY .EQ. 3 )  GO TO 30
C **  STATIC DEFLECTION
   10 CONTINUE
      WRITE (6,48) ( WORK1(I), I=1,NUVW )
   48 FORMAT ('1THE CONTRIBUTIONS OF THE SERIES TERMS TO DEFLECTION FOLL
     1OW'/(1X,10E12.4))
      CALL DISPLA ( WORK1, 1 )
      GO TO 1000
C **  FREE VIBRATION
   20 CONTINUE
      DO 9990 I=1,NUVW
      IF ( WORK1(I) .LE. .5 )  GO TO 9990
      ISTART = I
      GO TO 9991
 9990 CONTINUE
 9991 IFIN = ISTART + NMODES - 1
      DO 90 I=ISTART,IFIN
      WRITE (6,60) WORK1(I), MM(I), NN(I), ( T(I,J), J=1,NUVW )
   60 FORMAT ('1THE FREQUENCY IS ',E16.7,' CPS. FOR M = ',I2,', N = ',
     1 I2,'.'/'0THE CONTRIBUTIONS OF THE SERIES TERMS FOLLOW'/
     2 (1X,10E12.4))
      DO 70 J=1,NUVW
   70 WORK2(J) = T(I,J)
      CALL DISPLA  ( WORK2, 2 )
   90 CONTINUE
      GO TO 1000
C **  BUCKLING
   30 CONTINUE
      DO 200 I=1,IFLAGB
      BIG = 0.1
      NSAVE = 0
      DO 180 J=1,NW
      IF ( ABS (U(I,J)) .LE. BIG )  GO TO 180
      BIG = ABS (U(I,J))
      NSAVE = J
  180 WORK2(J) = U(I,J)
      M = ITX
```

```
      N = ITY
      IF ( NSAVE .EQ. 1 )  GO TO 6
      DO 5 J=2,NSAVE
      IF ( N+1-ITY .GE. NTWY )  GO TO 4
      N = N+1
      GO TO 5
    4 N = ITY
      M = M+1
    5 CONTINUE
    6 CONTINUE
      WRITE (6,190) WORK1(I), M, N, ( WORK2(J), J=1,NW )
  190 FORMAT ('0THE BUCKLING EIGENVALUE IS' E16.7,' FOR M ='I3,', N ='
     1         I3,'.'/'0THE CONTRIBUTIONS OF THE SERIES TERMS FOR W FOLL
     2OW'/(1X,10E12.4))
      CALL DISPLA  ( WORK2, 3 )
  200 CONTINUE
 1000 CONTINUE
      CALL STATUS ( ITIME )
      ET = .01*ITIME(8) - TIME(1)
      MINUTE = INT ( ET/60. )
      SEC = AMOD ( ET , 60. )
      ISEC = SEC
      WRITE (6,66) MINUTE, ISEC
   66 FORMAT ('0THE EXECUTION TIME FOR THIS PROBLEM WAS ',I3,' MINUTES,
     1',I2,' SECONDS.')
      RETURN
      END
```

CC = 00083

```
      SUBROUTINE STRESS
C **
C **  THIS SUBROUTINE CALCULATES STRESSES AND STRAINS.
C **
      DIMENSION    F(15,25,25),   FMAX(5)
      DIMENSION    A(3,3),        B(3,3),        D(3,3),        Z(41)
      DIMENSION    THETA(40),     THICK(40),     C11(40),       C22(40)
      DIMENSION    C12(40),       C66(40),       ANGCK(3,10),   MCHK(3)
      DIMENSION    EC(3,40),      ET(3,40),      SIG(5)
      DIMENSION    SIGS(5),       SMAR(5),       EPSN(5),       EPSS(5)
C
      COMMON / ARRAYS / F,  FMAX
      COMMON / ABD    / A,  B,  D,  RHAB,  THETA,  THICK,  C11,  C22,
     1                 C66,  C12,  EC,  ET,  ANGCK,  MCHK,  Z
      COMMON / CNTROL / I$(5),    IMATL,    J$(2),     IOUT
      COMMON / NUMBER / NPLYS
C
      DATA  X/'X'/, Y/'Y'/, YO/'LOW'/, UP/'UPP'/
      DATA  $X/'X-'/, $Y/'Y-'/, $Z /'XY'/, $1/'1-'/, $2/'2-'/, $3 /'12'/
      DATA  SIG(1)/'NORM'/, SIG(2)/'AL S'/, SIG(3)/'TRES'/
      DATA  SIG(4)/'SES '/, SIG(5)/' '/
      DATA  SIGS(1)/'SHEA'/, SIGS(2)/'R  S'/, SIGS(3)/'TRES'/
      DATA  SIGS(4)/'SES '/, SIGS(5)/' '/
      DATA  SMAR(1)/'MARG'/, SMAR(2)/'INS '/, SMAR(3)/'OF S'/
      DATA  SMAR(4)/'AFET'/, SMAR(5)/'Y'/
      DATA  EPSN(1)/'NORM'/, EPSN(2)/'AL S'/, EPSN(3)/'TRAI'/
      DATA  EPSN(4)/'NS  '/, EPSN(5)/' '/
      DATA  EPSS(1)/'SHEA'/, EPSS(2)/'R  S'/, EPSS(3)/'TRAI'/
      DATA  EPSS(4)/'NS  '/, EPSS(5)/' '/
C
      FMIN = 100.
      VAL = 1.
   10 FORMAT ('0',25F5.2)
      I = IOUT
      IF(I.EQ.4.OR.I.EQ.6.OR.I.EQ.7.OR.I.EQ.8)  GO TO 51
      WRITE (6,20)  X, FMAX(4)
      CALL  OUT ( 4)
   20 FORMAT ('1THE MIDDLE SURFACE STRAIN IN THE ',A1,' DIRECTION DIVIDE
     1D BY ',E15.6,'/10000 FOLLOWS')
      WRITE (6,20)  Y, FMAX(5)
      CALL  OUT ( 5)
      WRITE (6,30)  FMAX(6)
      CALL  OUT ( 6)
   30 FORMAT ('1THE MIDDLE SURFACE SHEAR STRAIN DIVIDED BY ',E15.6,
     1        '/10000 FOLLOWS')
      WRITE (6,40)  X, FMAX(7)
      CALL  OUT ( 7)
   40 FORMAT ('1THE CURVATURE IN THE ',A1,' DIRECTION DIVIDED BY ',
     1         E15.6,'/10000 FOLLOWS')
      WRITE (6,40)  Y, FMAX(8)
      CALL  OUT ( 8)
      WRITE (6,50)  FMAX(9)
      CALL  OUT ( 9)
   50 FORMAT ('1THE TWIST CURVATURE DIVIDED BY ',E15.6,'/10000 FOLLOWS')
      IF ( IOUT .EQ. 3 )  GO TO 999
   51 IF ( IMATL .EQ. 1 .OR. IMATL .EQ. 4 )  GO TO 150
```

```
C **  SOLID LAMINATE
      IF ( IOUT .LT. 7 )  GO TO 999
      NP = NPLYS + 1
      DO 100 N=1,NP
      ITEST = 0
      J = N
      IF ( Z(N) .GE. 0 )  J = N - 1
      IF ( C11(J) .LE. 10. )  GO TO 100
      DO 60 JJ=1,3
      J3 = JJ+3
      J6 = JJ+6
      DO 60 K=1,25
      DO 60 L=1,25
   60 F(JJ,K,L) = FMAX(J3)*F(J3,K,L) + Z(N)*FMAX(J6)*F(J6,K,L)
   70 IF ( ITEST .NE. 0 )  J = N
      ANG = THETA(J)
      CALL ROTATE ( 10, 1, ANG )
      CALL NRMLIZ ( 10, 12 )
      CALL MARGIN ( 10, 13, J )
      WRITE(6,80)J,THETA(J),EPSN,$1,FMAX(10)
      CALL  OUT (10)
      WRITE(6,80)J,THETA(J),EPSN,$2,FMAX(11)
      CALL  OUT (11)
      WRITE(6,80)J,THETA(J),EPSS,$3,FMAX(12)
      CALL  OUT (12)
      WRITE(6,80)J,THETA(J),SMAR,$1,VAL
      CALL  OUT (13)
      WRITE(6,80)J,THETA(J),SMAR,$2,VAL
      CALL  OUT (14)
      WRITE(6,80)J,THETA(J),SMAR,$3,VAL
      CALL  OUT (15)
   80 FORMAT ('1FOR LAYER ',I2,' ( THETA = ',F6.2,' ), THE ',4A4,A1, ' I
     1N THE ',A2,' DIRECTION DIVIDED BY ',E15.6,'/10000 FOLLOW')
      CALL SEARCH ( J,1,13,15,MH,KH,LH,IH,NH,FMIN )
      IF ( ABS(Z(N)) .GT. 1.E-4 )  GO TO 100
      IF ( ABS(THETA(N) - THETA(N-1)) .LT. .01 )  GO TO 100
      IF ( ITEST .EQ. 1 )  GO TO 100
      ITEST = 1
      GO TO 70
  100 CONTINUE
      IF ( MH .EQ. 13 )  $=$1
      IF ( MH .EQ. 14 )  $=$2
      IF ( MH .EQ. 15 )  $=$3
      WRITE (6,110)  $, IH, KH, LH, FMIN
  110 FORMAT ('1THE MINIMUM MARGIN OF SAFETY OCCURS FOR A STRAIN IN THE
     1',A2,' DIRECTION IN LAYER ',I2,' AT X = ',I2,', Y = ',I2,
     2        '.  ITS VALUE IS ',F5.2)
      GO TO 999
C **  ISOTROPIC OR SANDWICH
  150 CONTINUE
      DO 600 N=1,2
      SUR = YO
      IF (N.EQ.2) SUR = UP
      IF ( IMATL .EQ. 4 )  GO TO 160
      I = 1
      J = N
```

```
      GO TO 170
  160 I=1
      J=1
      IF (N.EQ.1)  GO TO 170
      I = 3
      J = 4
  170 CONTINUE
C **  CALCULATE COMBINED STRAINS IN PANEL AXES.
      DO 180 JJ=1,3
      J3 = JJ+3
      J6 = JJ+6
      DO 180 K=1,25
      DO 180 L=1,25
  180 F(JJ,K,L) = FMAX(J3)*F(J3,K,L) + Z(J) * FMAX(J6)*F(J6,K,L)
      IF ( IOUT .LT. 4 .OR. IOUT .EQ. 7 )  GO TO 240
C **  CALCULATE COMBINED STRESSES IN PANEL AXES.
      DO 190 K=1,25
      DO 190 L=1,25
      F(10,K,L) = C11(I) * F(1,K,L) + C12(I) * F(2,K,L)
      F(11,K,L) = C12(I) * F(1,K,L) + C22(I) * F(2,K,L)
  190 F(12,K,L) = C66(I) * F(3,K,L)
      CALL NRMLIZ ( 10, 12 )
      WRITE(6,200) SIG ,SUR,$X,FMAX(10)
      CALL  OUT (10)
      WRITE(6,200) SIG ,SUR,$Y,FMAX(11)
      CALL  OUT (11)
      WRITE(6,200) SIGS,SUR,$Z,FMAX(12)
      CALL  OUT (12)
  200 FORMAT ('1THE ',4A4,A1,' ON THE ',A3,'ER SURFACE IN THE ',A2,
     1         ' DIRECTION DIVIDED BY ',E15.6,'/10000 FOLLOW')
  240 CONTINUE
      IF ( IOUT .LT. 7 )  GO TO 600
      IF ( IMATL .EQ. 4 )  GO TO 400
C **  ISOTROPIC
      CALL NRMLIZ ( 1, 3 )
      CALL MARGIN ( 1, 10, I )
      WRITE(6,200) EPSN,SUR,$X,FMAX( 1)
      CALL  OUT ( 1)
      WRITE(6,200) EPSN,SUR,$Y,FMAX( 2)
      CALL  OUT ( 2)
      WRITE(6,200) EPSS,SUR,$Z,FMAX( 3)
      CALL  OUT ( 3)
      WRITE(6,200) SMAR,SUR,$X,VAL
      CALL  OUT (10)
      WRITE(6,200) SMAR,SUR,$Y,VAL
      CALL  OUT (11)
      WRITE(6,200) SMAR,SUR,$Z,VAL
      CALL  OUT (12)
      CALL SEARCH ( I,N,10,12,MH,KH,LH,IH,NH,FMIN )
      GO TO 600
C **  SANDWICH
  400 NCHK = MCHK(I)
      DO 500 NN=1,NCHK
      ANG = ANGCK(I,NN)
      CALL ROTATE ( 10, 1, ANG )
      CALL NRMLIZ ( 10, 12 )
```

```
      CALL MARGIN ( 10, 13, I )
      CALL SEARCH ( I,NN,13,15,MH,KH,LH,IH,NH,FMIN )
      WRITE(6,410) ANG,EPSN,SUR,$1,FMAX(10)
      CALL  OUT (10)
      WRITE(6,410) ANG,EPSN,SUR,$2,FMAX(11)
      CALL  OUT (11)
      WRITE(6,410) ANG,EPSS,SUR,$3,FMAX(12)
      CALL  OUT (12)
      WRITE(6,410) ANG,SMAR,SUR,$1,VAL
      CALL  OUT (13)
      WRITE(6,410) ANG,SMAR,SUR,$2,VAL
      CALL  OUT (14)
      WRITE(6,410) ANG,SMAR,SUR,$3,VAL
      CALL  OUT (15)
  410 FORMAT ('1FOR THETA = ',F6.2,', THE ',4A4,A1,' ON THE ',A3,
     1'ER SURFACE IN THE ',A2,' DIRECTION DIVIDED BY ',
     2 E15.6,'/10000 FOLLOW')
  500 CONTINUE
  600 CONTINUE
      IF ( IOUT .LT. 7 )  GO TO 999
      IF ( IMATL .EQ. 4 )  GO TO 620
      IF ( MH .EQ. 10 )  $=$1
      IF ( MH .EQ. 11 )  $=$2
      IF ( MH .EQ. 12 )  $=$3
      IF ( NH .EQ. 1 )  SUR = YO
      IF ( NH .EQ. 2 )  SUR = UP
      WRITE (6,610)  $, SUR, KH, LH, FMIN
  610 FORMAT ('1THE MINIMUM MARGIN OF SAFETY OCCURS FOR A STRAIN IN THE
     1',A2,' DIRECTION ON THE ',A3,'ER SURFACE AT X = ',I2,', Y = ',I2,
     2      '.'/' ITS VALUE IS ',F6.2)
      GO TO 999
  620 ANG = ANGCK(IH,NH)
      IF ( MH .EQ. 13 )  $=$1
      IF ( MH .EQ. 14 )  $=$2
      IF ( MH .EQ. 15 )  $=$3
      WRITE (6,630)  $, ANG, IH, KH, LH, FMIN
  630 FORMAT ('1THE MINIMUM MARGIN OF SAFETY OCCURS FOR A STRAIN IN THE
     1',A2,' DIRECTION AT AN ANGLE THETA OF ',F6.2,' DEGREES IN LAYER '
     2 ,I2,'.'/' IT IS LOCATED AT X = ',I2,', Y = ',I2,', AND HAS A VAL
     3UE OF ',F6.2)
  999 RETURN
      END
```

CC = 00210

```
      SUBROUTINE ROTATE  ( M, MX, ANG )                                 SS8U000
C **                                                                    SS8U001
C **  THIS SUBROUTINE PERFORMS A TRANSFORMATION OF COORDINATES          SS8U002
C **  FROM THETA = 0. TO THETA = ANG .                                  SS8U003
C **                                                                    SS8U004
      DIMENSION    F(15,25,25)                                          SS8U005
C                                                                       SS8U006
      COMMON / ARRAYS / F                                               SS8U007
C                                                                       SS8U008
      M1 = M+1                                                          SS8U009
      M2 = M+2                                                          SS8U010
      MX1 = MX+1                                                        SS8U011
      MX2 = MX+2                                                        SS8U012
      A = ANG * .0174533                                                SS8U013
      C  = COS(A)                                                       SS8U014
      S  = SIN(A)                                                       SS8U015
      C2 = C*C                                                          SS8U016
      S2 = S*S                                                          SS8U017
      SC = S*C                                                          SS8U018
      DO 10 K=1,25                                                      SS8U019
      DO 10 L=1,25                                                      SS8U020
      F(M,K,L) = F(MX,K,L)*C2 + F(MX1,K,L)*S2 + F(MX2,K,L)*SC           SS8U021
      F(M1,K,L) = F(MX,K,L)*S2 + F(MX1,K,L)*C2 + F(MX2,K,L)*SC          SS8U022
   10 F(M2,K,L) = -2.*SC*( F(MX,K,L) - F(MX1,K,L) ) + F(MX2,K,L)*(C2-S2)SS8U023
      RETURN                                                            SS8U024
      END                                                               SS8U025

                                                      CC = 00026
```

```
      SUBROUTINE NRMLIZ ( M1, M2 )
C **
C **  THE INPUT ARRAYS ARE NORMALIZED BY THEIR LARGEST VALUES.
C **
      DIMENSION    F(15,25,25),   FMAX(15)
C
      COMMON / ARRAYS / F,        FMAX
C
      DO 30 M=M1,M2
      FMAX(M) = F(M,1,1)
      DO 10 K=1,25
      DO 10 L=1,25
      FD = ABS ( F(M,K,L) )
      IF ( FD .GT. FMAX(M) )  FMAX(M) = FD
   10 CONTINUE
      IF ( ABS ( FMAX(M) ) .LT. 1.E-10 )  FMAX(M) = 1.
      DO 20 K=1,25
      DO 20 L=1,25
   20 F(M,K,L) = F(M,K,L) / FMAX(M)
   30 CONTINUE
      RETURN
      END
```

CC = 00022

```
      SUBROUTINE MARGIN ( MSTRN, MMAR, LAY )                             SS8W000
C **                                                                     SS8W001
C **  THIS SUBROUTINE CALCULATES MARGINS OF SAFETY ACCORDING             SS8W002
C **  TO THE MAXIMUM STRAIN THEORY.                                      SS8W003
C **                                                                     SS8W004
      DIMENSION    F(15,25,25),   FMAX(15), EA(3), ET(3,40), EC(3,40)   SS8W005
C                                                                        SS8W006
      COMMON / ARRAYS / F,        FMAX                                   SS8W007
      COMMON / ABD    / DUM(268),       EC,        ET                    SS8W008
C                                                                        SS8W009
      DO 10 M=1,3                                                        SS8W010
      I= M+MSTRN -1                                                      SS8W011
      J= M+MMAR -1                                                       SS8W012
      DO 10 K=1,25                                                       SS8W013
      DO 10 L=1,25                                                       SS8W014
      EA(M) = ET(M,LAY)                                                  SS8W015
      IF ( F(I,K,L) .LE. 0. )  EA(M) = EC(M,LAY)                         SS8W016
      F(J,K,L) = 9.0                                                     SS8W017
      IF ( F(I,K,L) .NE. 0. )  F(J,K,L) = EA(M)/F(I,K,L)/FMAX(I) - 1.    SS8W018
      IF ( F(J,K,L) .GE. 9.99 )  F(J,K,L) = 9.98                         SS8W019
      IF ( F(J,K,L) .LE.-9.99 )  F(J,K,L) =-9.98                         SS8W020
   10 CONTINUE                                                           SS8W021
      RETURN                                                             SS8W022
      END                                                                SS8W023

                                                      CC = 00024
```

```
      SUBROUTINE REDUCE ( NOPT,V,Z1,Z2,Z3,Z4,WORK1,WORK2,NUV,NW )
      DIMENSION V(150,150), Z1(100,100), Z2(100,50), Z3(50,100),
     1          Z4(50,50), WORK1(150), WORK2(150)
C
      DO 10 I=1,NUV
      DO 10 J=1,NUV
   10 Z1(I,J) = V(I,J)
      DO 20 I=1,NUV
      DO 20 J=1,NW
   20 Z2(I,J) = V(I,J+NUV)
      DO 30 I=1,NW
      DO 30 J=1,NUV
   30 Z3(I,J) = V(I+NUV,J)
      CALL GJINV ( Z1,NUV,0,IER,WORK1,WORK2,100 )
      CALL SWITCH ( Z1, NUV, 50, 1., 0. )
      CALL YOSFEM ( 2,Z1,NUV,NUV,50,Z2,NW,50,V,WORK1 )
      CALL YOSFEM ( 3,Z3,NW,NUV,25,Z2,NW,50,Z4,WORK1 )
      DO 40 I=1,NW
      DO 40 J=1,NW
   40 Z4(I,J) = V(I+NUV,J+NUV) - Z4(I,J)
   50 DO 60 I=1,NW
      DO 60 J=1,NW
   60 V(I,J) = Z4(I,J)
  999 RETURN
      END
```

CC = 00025

```
      SUBROUTINE FLEX
C     THIS SUBROUTINE CALCULATES THE FLEXIBILITY MATRIX AT THE
C     DESIRED POINTS.
      COMMON / FLEXBL / XP(50),    YP(50)
      COMMON / ZWORK  / W(50,50),       EM(50,50),      FL(50,50),
     1                  W1(50),        W2(50)
      COMMON / VALUES / E(4,2,3,10,25)
      COMMON / NUMBER / NPLYS, NTX, N$(2), NTY, M$(15), MAT, NUV, NW
      COMMON / CNTROL / I$(14), IFLEX
      COMMON / ARRAYS / V(150,150)
      DO 10 I=1,NW
      DO 10 J=1,NW
   10 W(I,J) = V(I,J)
      CALL GJINV ( W, NW, 0, IER, W1, W2, 50 )
      DO 20 II=1,IFLEX
      I = XP(II)*24 + 1
      J = YP(II)*24 + 1
      IF ( I.LT.1 ) I=1
      IF ( J.LT.1 ) J=1
      IF ( I.GT.24) I=24
      IF ( J.GT.24) J=24
      IP1 = I + 1
      JP1 = J + 1
      DELX = XP(II)*24. - (I-1)
      DELY = YP(II)*24. - (J-1)
      DO 20 L = 1,NTX
      DO 20 K = 1,NTY
      JJ = NTY*(L-1) + K
      EVX = E(1,1,3,L,I)*(1.-DELX) + E(1,1,3,L,IP1)*DELX
      EVY = E(1,2,3,K,J)*(1.-DELY) + E(1,2,3,K,JP1)*DELY
      EM(II,JJ) = EVX * EVY
   20 CONTINUE
      DO 60 II=1,IFLEX
      DO 40 JJ=1,NW
      W1(JJ) = 0.
      DO 30 KK=1,NW
   30 W1(JJ) = W1(JJ) + W(JJ,KK) * EM(II,KK)
   40 CONTINUE
      DO 50 LL=1,IFLEX
      FL(II,LL) = 0.
      DO 50 KK=1,NW
   50 FL(II,LL) = FL(II,LL) + EM(LL,KK) * W1(KK)
   60 CONTINUE
      WRITE (6,70)
   70 FORMAT ('1FLEXIBILITY MATRIX')
      DO 90 I=1,IFLEX
      WRITE (6,80) I, ( FL(I,J), J=1,IFLEX )
   80 FORMAT ('0ROW',I3//(1P6E16.6))
   90 CONTINUE
      RETURN
      END
```

CC = 00051

```
      SUBROUTINE KDF ( BUCKNX )
C
C     COMPUTES AXIAL BUCKLING NX FOR IMPERFECT ANISOTROPIC CYLINDERS
C
      DIMENSION    AS(3,3),  BS(3,3),  DS(3,3),  W1(3),    W2(3)
      COMMON / ABD /  A(3,3),  B(3,3),  D(3,3)
      COMMON / GEOM /  AA, BB, RR,  S$(4),  MU
      COMMON / CUBE /  P1, P2, P3, P4, ROOT
      DIMENSION  ITIME(12)
      DIMENSION  ATAU(20), AMDA(20)
      REAL  MU
C
      FAC = 100
      RHO = .707
      DO 10 I=1,3
      DO 10 J=1,3
   10 AS(I,J) = A(I,J)
      CALL GJINV ( AS, 3, 0, IER, W1, W2, 3 )
C **  AS = A**-1
      DO 20 I=1,3
      DO 20 J=1,3
   20 BS(I,J) = - B(I,J)
      CALL YOSFEM ( 2, AS, 3, 3, 3, BS, 3, 3, D, W1 )
C     BS = - A**-1 * B
      CALL YOSFEM ( 3, B, 3, 3, 3, BS, 3, 3, DS, W1 )
      DO 30 I=1,3
      DO 30 J=1,3
   30 DS(I,J) = D(I,J) + DS(I,J)
C     DS = D - B * A**-1 * B
      GAM = 1./SQRT(AS(2,2)*DS(1,1))
      ALP = DS(1,1)*GAM
      BET = BS(2,1)*GAM
      IMAX = 10
      TAU0 = 0.
      FTAU = 10.
   40 DO 100 I=1,IMAX
      ATAU(I) = TAU0 + I/FTAU
      TAU = ATAU(I)
      D12 = DS(1,1)*RHO**4 + ( 2.*DS(1,2) + 4.*DS(3,3) ) *RHO**2*TAU**2
     1    + DS(2,2)*TAU**4
      A11 = AS(2,2)*RHO**4 + ( 2.*AS(1,2) + AS(3,3) ) *RHO**2*TAU**2
     1    + AS(1,1)*TAU**4
      A13 = AS(2,2)*81.*RHO**4 + ( 2.*AS(1,2) + AS(3,3) )*9.*RHO**2
     1    * TAU**2 + AS(1,1)*TAU**4
      A21 = -2.*AS(2,3)*RHO**3*TAU - 2.*AS(1,3)*RHO*TAU**3
      A22 = -A21
      A23 = 2.*AS(2,3)*27.*RHO**3*TAU + 2.*AS(1,3)*3.*RHO*TAU**3
      B11 = BS(2,1)*RHO**4 + ( BS(1,1) + BS(2,2) - 2.*BS(3,3) )*RHO**2
     1      * TAU**2 + BS(1,2)*TAU**4
      B11P = B11 - 2.*RHO*RHO/GAM
      B22 = ( BS(3,1) - 2.*BS(2,3) ) * RHO**3*TAU + ( BS(3,2)
     1      - 2.*BS(1,3) ) * RHO*TAU**3
      C1 = RHO*RHO + (1.-2.*RHO*RHO*BET)**2/4./RHO/RHO
      D1 = A11*A11 - A21*A21
      D3 = A13*A13 - A23*A23
      A1 = D12 + ( (A11*B11P - A22*B22) * B11P + (A11*B22 - A22*B11P)
```

```
     1     * B22 ) / D1
      A2 = 4.*ALP*RHO*RHO/GAM
      A3 = 4.*MU*RHO*RHO*TAU*TAU* (A11*B11P - A22*B22) *C1/D1
      A4 =    MU*ALP* (1. - 2.*RHO*RHO*BET)        *TAU*TAU
      A5 = 4.*MU*MU*RHO**4*TAU**4*C1*C1* (A11/D1 + A13/D3 )
      P1 = A2
      P2 = - (A1 + 2.*A2*C1 + A4)
      P3 = 2.*A1*C1 + A2*C1*C1 + A4*C1 + A3
      P4 = - (A1*C1*C1 + A3*C1 + A5)
      CALL CUBIC
      AMDA(I) = ROOT
  100 CONTINUE
      CALL MIN ( AMDA, IMAX, IMIN )
      IF ( IMAX .EQ. 20 )  GO TO 200
      TAU0 = ATAU(IMIN) - .1
      IMAX = 20
      FTAU = 100
      GO TO 40
  200 CONTINUE
      FAC = AMDA(IMIN)
      TAU = ATAU(IMIN)
   50 BUCKNX = 2.*ALP*FAC/RR
      PBUCK = 2.*RR*BUCKNX*3.14159
      WRITE (6,600)  RHO, TAU, FAC, BUCKNX, PBUCK
  600 FORMAT ('OIMPERFECTION SENSITIVITY ANALYSIS FOR FULL CYLINDER  --
     1 RHO, TAU, LAMBDA-CR, NX-CR, P-CR'//' ',5E20.6)
      RETURN
      END
```

CC = 00084

```
      SUBROUTINE CUBIC
C     SOLVES A CUBIC POLYNOMIAL FOR THE REAL ROOT BY NEWTON-RAPHSON
      COMMON / CUBE / P1, P2, P3, P4, Y
      X = 1
      I = 0
    1 F = P1*X*X*X + P2*X*X + P3*X + P4
      I = I + 1
      FP = 3.*P1*X*X + 2.*P2*X + P3
      Y = X - F/FP
      IF (ABS(1-Y/X).LE. .001 ) GO TO 10
      X = Y
      IF ( I .LT. 10 )  GO TO 1
   10 CONTINUE
      A = P1
      B = P1*Y + P2
      C = P1*Y*Y + P2*Y + P3
      DISC = B*B - 4.*A*C
      IF (DISC) 20,30,30
   20 WRITE (6,70)
   70 FORMAT ('0OTHER TWO ROOTS ARE COMPLEX')
      GO TO 100
   30 X1 = ( -B + SQRT(DISC) ) /2./A
      X2 = ( -B - SQRT(DISC) ) /2./A
      Y = AMIN1 ( Y,X1,X2 )
  100 CONTINUE
      RETURN
      END
```

CC = 00027

```
      SUBROUTINE  OUT  ( N )
C **
C **  THIS SUBROUTINE PUTS THE ARRAYS OF OUTPUT IN A FORM FOR
C **        EFFICIENT WRITING.
C **
      COMMON / ARRAYS / F(15,25,25), FMAX(15), LIST(625)
C
      DO 10 K=1,25
      DO 10 L=1,25
      J = ( K-1 ) * 25 + L
   10 LIST(J) = F(N,K,L) * 10000
      WRITE (6,20) LIST
   20 FORMAT ('0',25I5)
      RETURN
      END
```

CC = 00015

```
      SUBROUTINE MIN ( VEC, N, IMIN )
      DIMENSION VEC(N)
      SMALL = 10.
      DO 10 I=1,N
      IF ( SMALL .LT. VEC(I) )  GO TO 10
      SMALL = VEC(I)
      IMIN = I
   10 CONTINUE
      RETURN
      END
```

CC = 00010